**이름 –** 이상호

**학번 –** 12212312

**학과 –** 경영학과

**데이터 –** [Default of Credit Card Clients Dataset | Kaggle](https://www.kaggle.com/datasets/uciml/default-of-credit-card-clients-dataset)

**분석 목표 –** 연령, 성별, 교육 수준, 결혼 여부에 따라 신용카드 사용액, 상환 여부, 한도 등이 어떻게 달라지는 지를 분석하고자 한다.

**분석 내용 –** 이 데이터는 대만의 어떤 신용카드사에서 고객들의 신용카드 사용 정보, 신상 정보들을 모아놓은 것으로 2005년 4월부터 9월까지의 데이터가 존재한다. 이 데이터는 다음과 같은 변수로 이루어져 있다.

ID: ID of each client

LIMIT\_BAL: Amount of given credit in NT dollars (includes individual and family/supplementary credit)

SEX: Gender (1=male, 2=female)

EDUCATION: (1=graduate school, 2=university, 3=high school, 4=others, 5=unknown, 6=unknown)

MARRIAGE: Marital status (1=married, 2=single, 3=others)

AGE: Age in years

PAY\_0: Repayment status in September 2005 (-1=pay duly, 1=payment delay for one month, 2=payment delay for two months, … 8=payment delay for eight months, 9=payment delay for nine months and above)

PAY\_2: Repayment status in August 2005 (scale same as above)

PAY\_3: Repayment status in July 2005 (scale same as above)

PAY\_4: Repayment status in June 2005 (scale same as above)

PAY\_5: Repayment status in May 2005 (scale same as above)

PAY\_6: Repayment status in April 2005 (scale same as above)

BILL\_AMT1: Amount of bill statement in September 2005 (NT dollar)

BILL\_AMT2: Amount of bill statement in August 2005 (NT dollar)

BILL\_AMT3: Amount of bill statement in July 2005 (NT dollar)

BILL\_AMT4: Amount of bill statement in June 2005 (NT dollar)

BILL\_AMT5: Amount of bill statement in May 2005 (NT dollar)

BILL\_AMT6: Amount of bill statement in April 2005 (NT dollar)

PAY\_AMT1: Amount of previous payment in September 2005 (NT dollar)

PAY\_AMT2: Amount of previous payment in August 2005 (NT dollar)

PAY\_AMT3: Amount of previous payment in July 2005 (NT dollar)

PAY\_AMT4: Amount of previous payment in June 2005 (NT dollar)

PAY\_AMT5: Amount of previous payment in May 2005 (NT dollar)

PAY\_AMT6: Amount of previous payment in April 2005 (NT dollar)

default.payment.next.month: Default payment (1=yes, 0=no)

우선 모든 변수에 값이 존재하므로 결측치를 채울 필요는 없다.

print(dataframe.isnull().sum())

>>> LIMIT\_BAL 0

SEX 0

EDUCATION 0

MARRIAGE 0

AGE 0

PAY\_0 0

PAY\_2 0

PAY\_3 0

PAY\_4 0

PAY\_5 0

PAY\_6 0

BILL\_AMT1 0

BILL\_AMT2 0

BILL\_AMT3 0

BILL\_AMT4 0

BILL\_AMT5 0

BILL\_AMT6 0

PAY\_AMT1 0

PAY\_AMT2 0

PAY\_AMT3 0

PAY\_AMT4 0

PAY\_AMT5 0

PAY\_AMT6 0

default.payment.next.month 0

그리고 시각화를 위해서는 변수 이름을 알아보기 쉽게 만드는 것이 좋다. 우선 해당 월의 카드 대금 상환 현황을 나타내는 PAY\_ 변수의 이름을 REPAYMENT STATUS\_ 로 바꾼다.

dataframe.rename(columns = {'PAY\_0' : 'REPAYMENT STATUS\_SEP', 'PAY\_2' : 'REPAYMENT STATUS\_AUG', 'PAY\_3' : 'REPAYMENT STATUS\_JUL'  
 , 'PAY\_4' : 'REPAYMENT STATUS\_JUN', 'PAY\_5' : 'REPAYMENT STATUS\_MAY', 'PAY\_6' : 'REPAYMENT STATUS\_APR'}, inplace = True)

해당 월의 신용카드 청구액을 나타내는 BILL\_AMT 변수의 이름은 CARD BILL\_ 로 바꾼다.

dataframe.rename(columns = {'BILL\_AMT1' : 'CARD BILL\_SEP', 'BILL\_AMT2' : 'CARD BILL\_AUG', 'BILL\_AMT3' : 'CARD BILL\_JUL'  
 , 'BILL\_AMT4' : 'CARD BILL\_JUN', 'BILL\_AMT5' : 'CARD BILL\_MAY', 'BILL\_AMT6' : 'CARD BILL\_APR'}, inplace = True)

해당 월의 선불결제 이용액을 나타내는 PAY\_AMT 변수의 이름을 PREPAID\_ 로 바꾼다.

dataframe.rename(columns = {'PAY\_AMT1' : 'PREPAID\_SEP', 'PAY\_AMT2' : 'PREPAID\_AUG', 'PAY\_AMT3' : 'PREPAID\_JUL'  
 , 'PAY\_AMT4' : 'PREPAID\_JUN', 'PAY\_AMT5' : 'PREPAID\_MAY', 'PAY\_AMT6' : 'PREPAID\_APR'}, inplace = True)

마지막으로 고객의 신용한도를 나타내는 LIMIT\_BAL변수의 이름을 CREDIT LIMIT으로, 고객의 현재 체납 여부를 나타내는 default.payment.next.month 변수의 이름을 DEFAULT 로 바꾼다.

dataframe.rename(columns = {'LIMIT\_BAL' : 'CREDIT LIMIT', 'default.payment.next.month' : 'DEFAULT'}, inplace = True)

또한 SEX, EDUCATION, MARRIAGE, REPAYMENT STATUS\_ 등은 숫자로 표시되어있지만 사실은 범주 변수이므로 그에 맞게 관측값의 이름도 바꾼다.

dataframe.replace({'SEX' : 1}, 'Male', inplace = True)  
dataframe.replace({'SEX' : 2}, 'Female', inplace = True)  
dataframe.replace({'EDUCATION' : 1}, 'Graduate', inplace = True)  
dataframe.replace({'EDUCATION' : 2}, 'Undergraduate', inplace = True)  
dataframe.replace({'EDUCATION' : 3}, 'High School', inplace = True)  
dataframe.replace({'EDUCATION' : 4}, 'Less than Middle School', inplace = True)  
dataframe.replace({'EDUCATION' : 5}, 'Unknown', inplace = True)  
dataframe.replace({'EDUCATION' : 6}, 'Unknown', inplace = True)  
dataframe.replace({'MARRIAGE' : 1}, 'Married', inplace = True)  
dataframe.replace({'MARRIAGE' : 2}, 'Single', inplace = True)  
dataframe.replace({'MARRIAGE' : 3}, 'Others', inplace = True)  
RepayStatVariable = ['REPAYMENT STATUS\_SEP', 'REPAYMENT STATUS\_AUG', 'REPAYMENT STATUS\_JUL', 'REPAYMENT STATUS\_JUN', 'REPAYMENT STATUS\_MAY', 'REPAYMENT STATUS\_APR']  
for i in RepayStatVariable:  
 dataframe.replace({i : -1}, 'No Overdue', inplace=True)  
 dataframe.replace({i : 1}, 'Short-term Overdue', inplace = True)  
 dataframe.replace({i : 2}, '2 Months ~ 6 Months Overdue', inplace = True)  
 dataframe.replace({i : 3}, '2 Months ~ 6 Months Overdue', inplace = True)  
 dataframe.replace({i : 4}, '2 Months ~ 6 Months Overdue', inplace = True)  
 dataframe.replace({i : 5}, '2 Months ~ 6 Months Overdue', inplace = True)  
 dataframe.replace({i : 6}, 'Long-term Overdue', inplace = True)  
 dataframe.replace({i : 7}, 'Long-term Overdue', inplace=True)  
 dataframe.replace({i : 8}, 'Long-term Overdue', inplace=True)  
 dataframe.replace({i : 9}, 'Long-term Overdue', inplace=True)

dataframe.replace({'DEFAULT': 0}, 'Yes', inplace=True)  
dataframe.replace({'DEFAULT': 1}, 'No', inplace=True)

학력을 알 수 없는 고객과 결혼상태가 Others인 고객은 분석에 적합하지 않으므로 삭제한다.

또 관측값이 0으로 되어있는 경우에는 사실상 결측치이므로 삭제한다.

UnkownCustomer = dataframe.loc[(dataframe['EDUCATION'] == 'Unknown') | (dataframe['EDUCATION'] == 0) | (dataframe['MARRIAGE'] == 'Others') | (dataframe['MARRIAGE'] == 0)].index  
dataframe.drop(UnkownCustomer, inplace = True)

REPAYMENT STATUS\_ 에서 0과 -2는 데이터 범례에서 확인할 수 없었다. 보통 연체를 하지 않는 이용자가 대부분인 경우가 많은 점, 0개월 연체나 -2개월 연체는 불가능 하다는 점을 고려해 둘다 모두 연체가 없다고 기록한다.

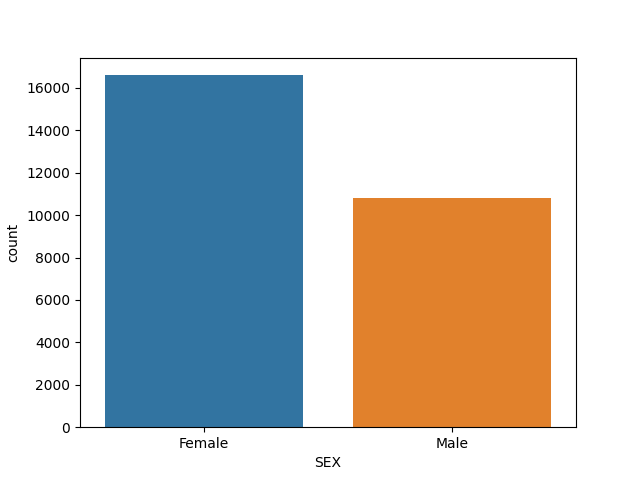
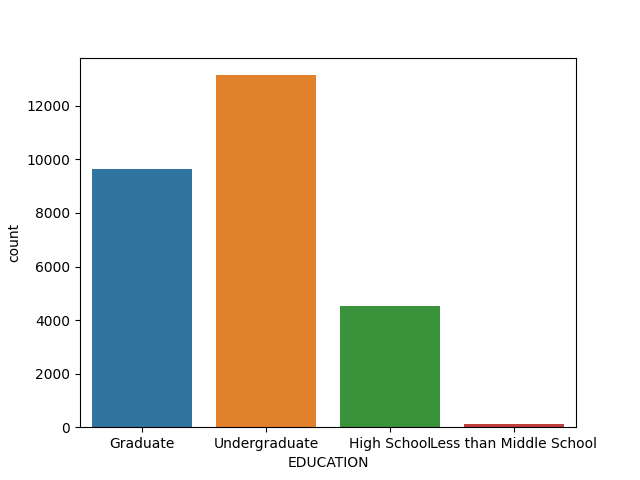
RepayStatVariable = ['REPAYMENT STATUS\_SEP', 'REPAYMENT STATUS\_AUG', 'REPAYMENT STATUS\_JUL', 'REPAYMENT STATUS\_JUN', 'REPAYMENT STATUS\_MAY', 'REPAYMENT STATUS\_APR']  
for i in RepayStatVariable:  
 dataframe.replace({i : 0}, 'No Overdue', inplace=True)  
 dataframe.replace({i : -2}, 'No Overdue', inplace = True)

CARD BILL\_ 이 0 미만인 경우는 상식적으로 설명이 불가능하므로 해당 고객은 삭제한다.

MinersBillCustomer = dataframe.loc[(dataframe['CARD BILL\_SEP'] < 0) | (dataframe['CARD BILL\_AUG'] < 0) | (dataframe['CARD BILL\_JUL'] < 0) |  
 (dataframe['CARD BILL\_JUN'] < 0) | (dataframe['CARD BILL\_MAY'] < 0) | (dataframe['CARD BILL\_APR'] < 0)].index  
dataframe.drop(MinersBillCustomer, inplace = True)

이상으로 데이터 전처리는 끝났다. 이제 어떤 신용카드사에서 고객의 인구통계학적 정보에 따라 신용카드 청구액, 선불결제 이용액, 대금 상환 현황, 채무불이행 가능성 등이 어떻게 다 른지를 분석해본다고 가정하고 데이터를 시각화 해보자

우선 표본의 특성을 간단하게 알아보자.



print(dataframe['SEX'].value\_counts())

>>> Female 16599

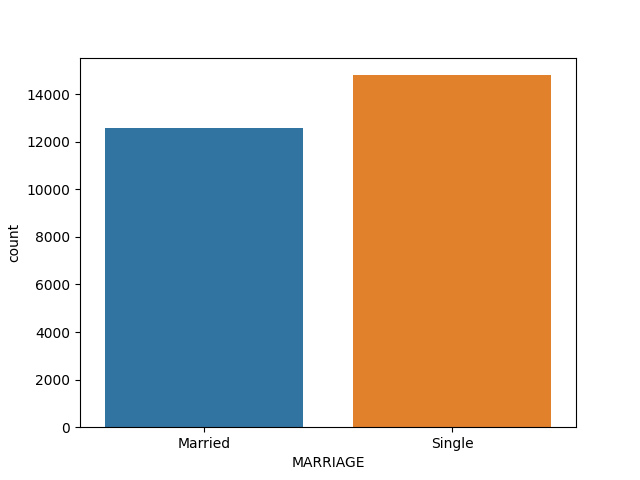
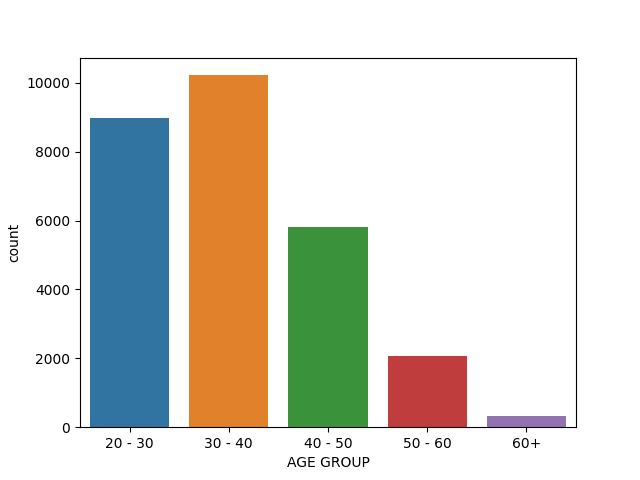
Male 10798

print(dataframe['EDUCATION'].value\_counts())

>>> Undergraduate 13144 Less than Middle School 107

Graduate 9638 High School 4508

성비는 여성이 약 60.5% 남성이 약 39.5%로 여성이 많았다. 또 학력에 있어서는 대학생이 약 48%로 가장 많았고, 대졸 약 35%, 고졸 약 16% 순으로 중졸 이하는 거의 없었다.

print(dataframe['MARRIAGE'].value\_counts())

>>> Single 14798

Married 12599

dataframe.loc[dataframe[(dataframe['AGE'] >= 20) & (dataframe['AGE'] < 30)].index, 'AGE GROUP'] = '20 - 30'  
dataframe.loc[dataframe[(dataframe['AGE'] >= 30) & (dataframe['AGE'] < 40)].index, 'AGE GROUP'] = '30 - 40'  
dataframe.loc[dataframe[(dataframe['AGE'] >= 40) & (dataframe['AGE'] < 50)].index, 'AGE GROUP'] = '40 - 50'  
dataframe.loc[dataframe[(dataframe['AGE'] >= 50) & (dataframe['AGE'] < 60)].index, 'AGE GROUP'] = '50 - 60'  
dataframe.loc[dataframe[dataframe['AGE'] >= 60].index, 'AGE GROUP'] = '60+'

print(dataframe['AGE GROUP'].value\_counts())

>>> 20 - 30 8982

30 - 40 10219

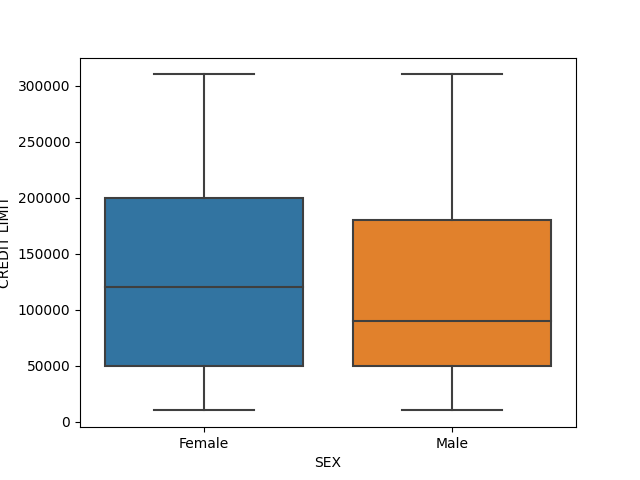
40 - 50 5812

50 - 60 2072

60+ 312

결혼 여부는 미혼 약 54%, 기혼 약 46%로 미혼이 약간 더 많았다. 나이는 범주 변수인 나이 대 변수로 바꾸어 시각화했다. 그 결과 20대 약 32%, 30대 약 37%, 40대 약 21%, 50대 약 7%로 20대와 30대가 가장 많았고 연령대가 올라갈수록 비율이 줄어들었다.

인구통계학적 변수에는 성별, 학력, 결혼여부, 나이대의 4개의 변수가 존재한다. 먼저 성별을 기준으로 변수들을 분석해보자



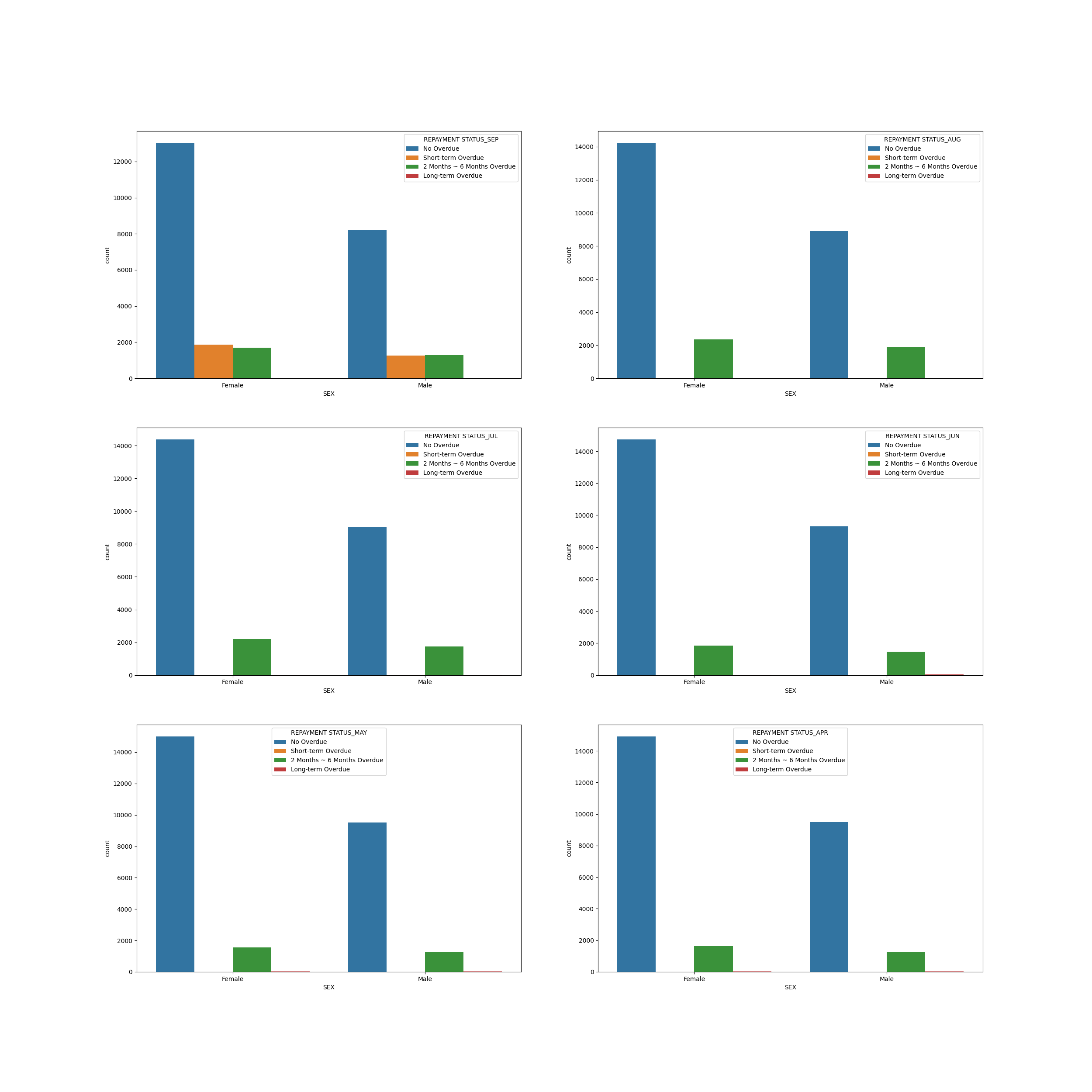
Q1\_cl = dataframe['CREDIT LIMIT'].quantile(0.25)  
Q3\_cl = dataframe['CREDIT LIMIT'].quantile(0.75)  
IQR\_cl = Q3\_cl - Q1\_cl  
dataframe\_CL = dataframe.loc[(dataframe['CREDIT LIMIT'] > Q1\_cl - 0.5 \* IQR\_cl) & (dataframe['CREDIT LIMIT'] < Q3\_cl + 0.5 \* IQR\_cl)]

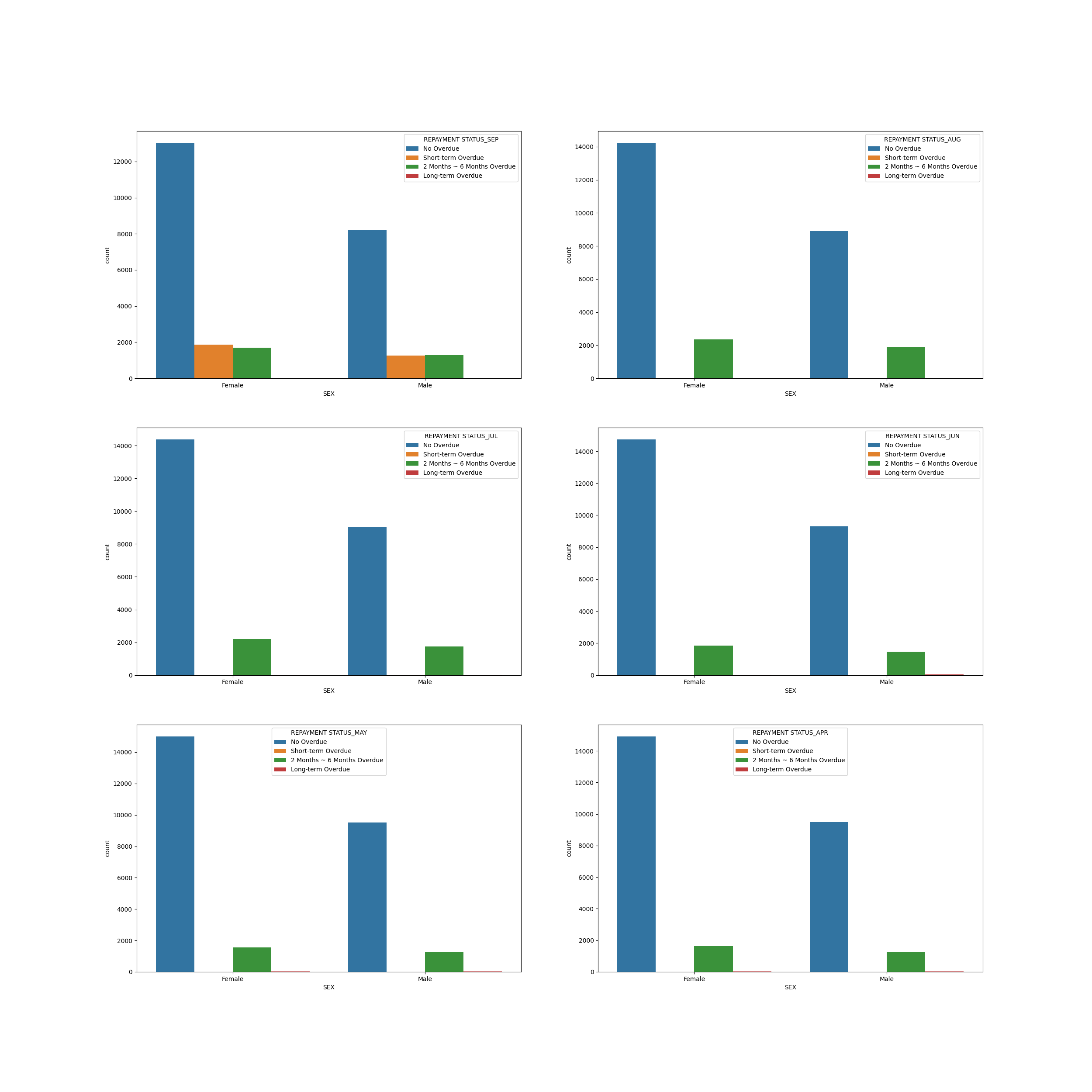
print(dataframe.groupby(['SEX'])['CREDIT LIMIT'].median())

>>> Female 140000.0

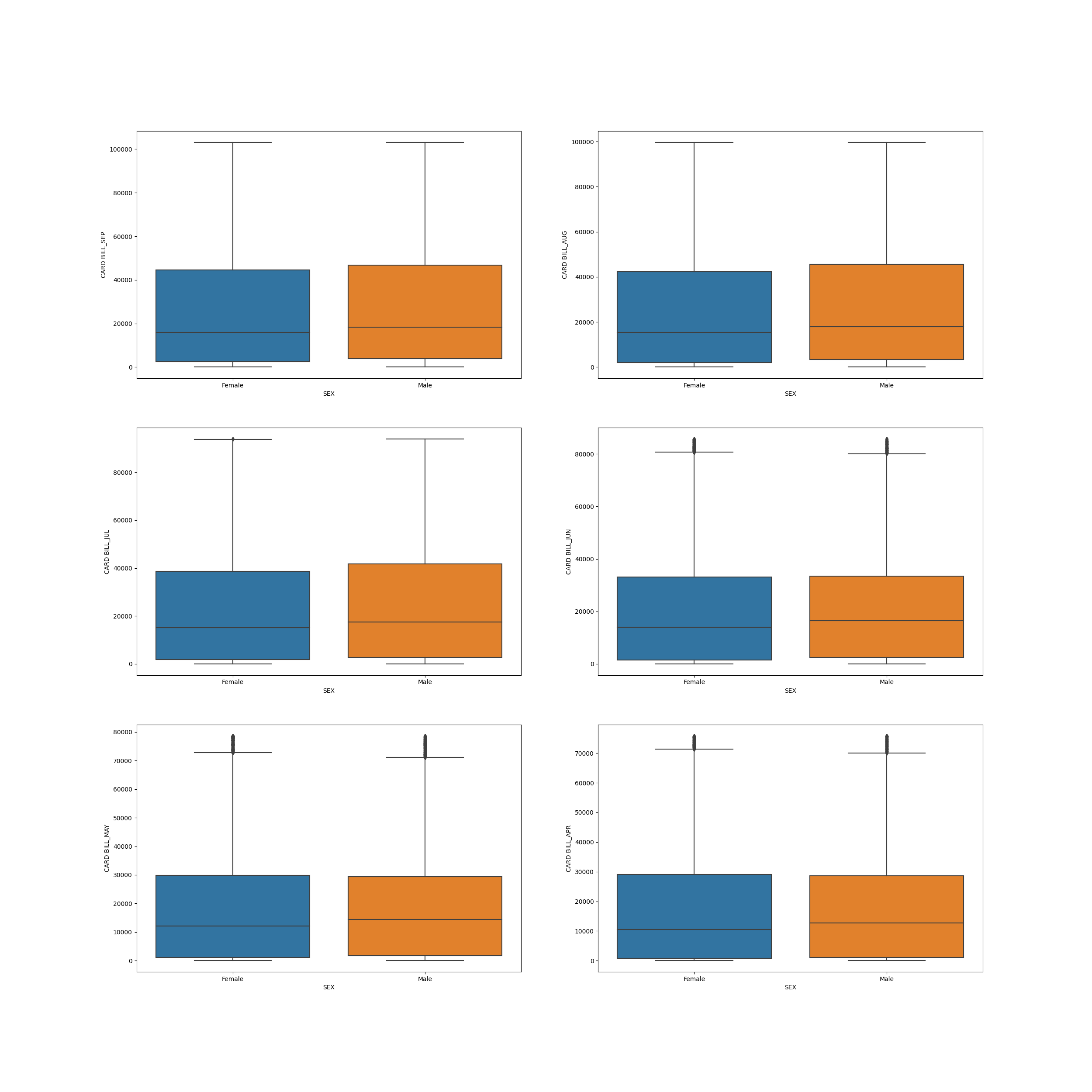
Male 120000.0

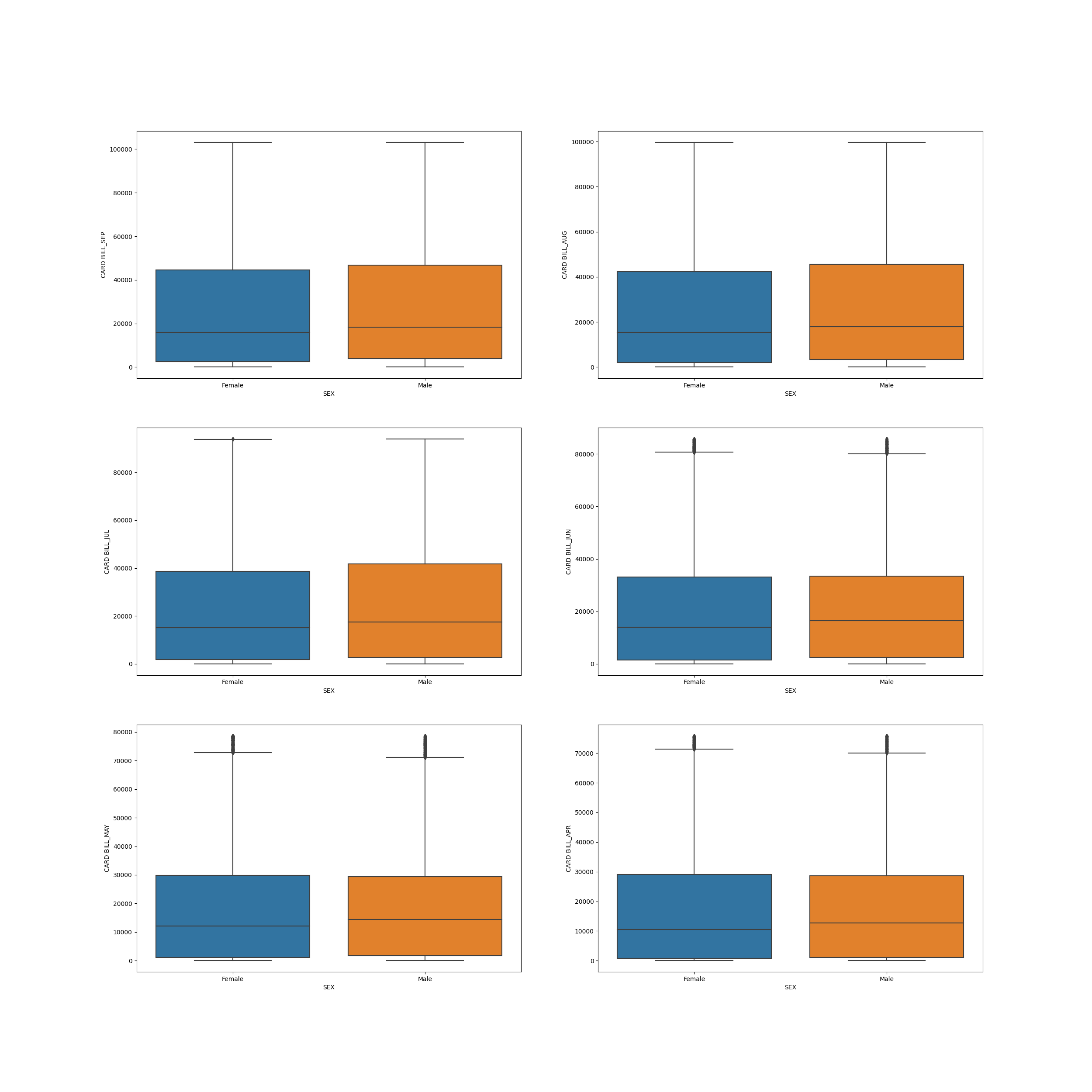
신용한도 변수에 처리를 하지 않은 상태에서는 이상치가 매우 많았다. 따라서 이상치가 보이 지 않도록 관측값의 구간을 조절했다. 이상치를 포함했을 때 신용한도 중앙값은 여성이 2만 신대만달러만큼 즉, 약 16.6% 만큼 높았다.





상환 현황에 있어서는 성별에 따라 차이가 크지 않았다. 비율상의 차이는 약간 있으나 남녀 불문 연체가 없는 고객이 다수였으며 최대 약 13% ~ 15% 정도의 고객은 2개월에서 6개월 정도의 연체가 있었다. 1개월 미만의 단기연체나 6개월 이상의 장기연체의 경우는 9월을 제 외하면 거의 없었다.





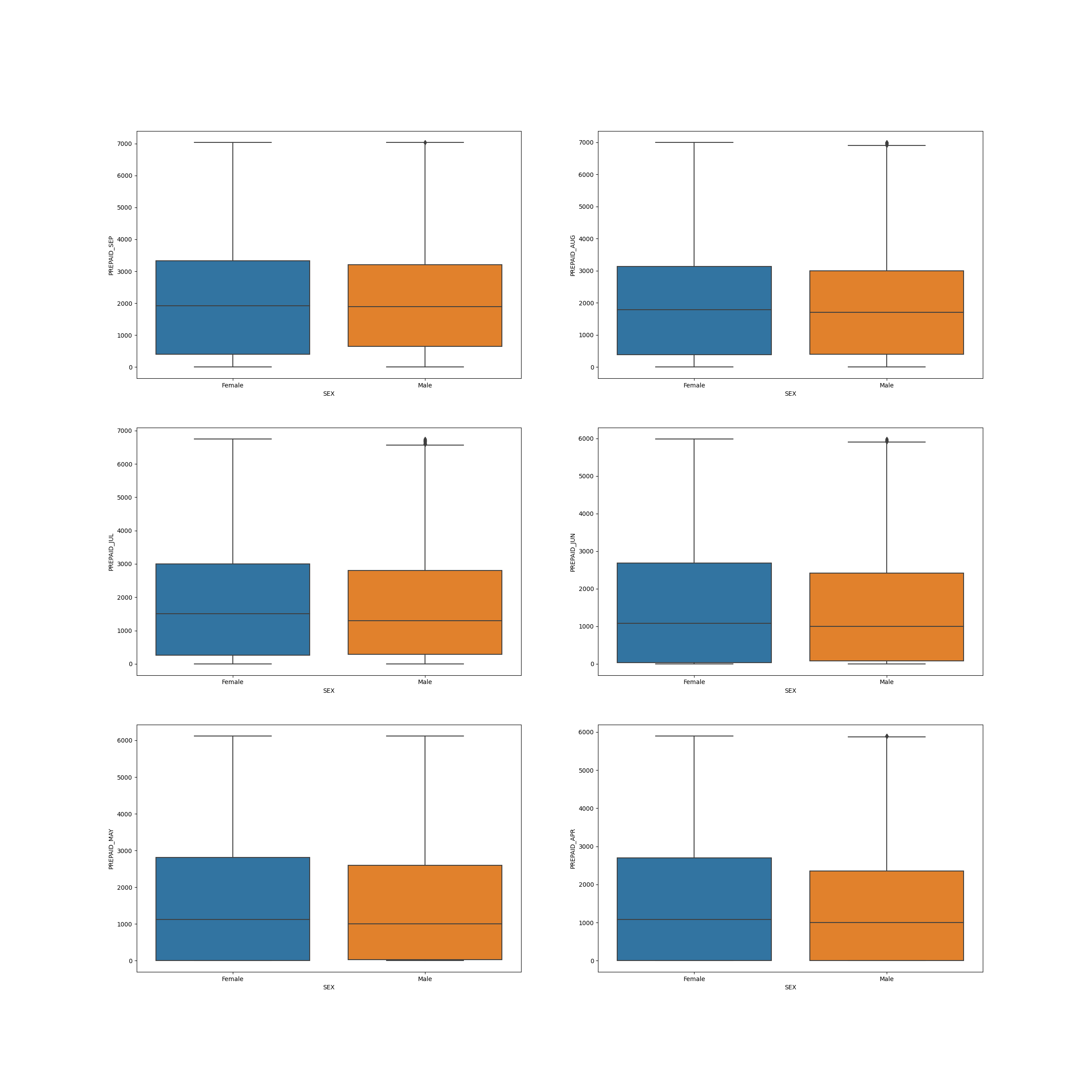
CardBillVariable = ['CARD BILL\_SEP', 'CARD BILL\_AUG', 'CARD BILL\_JUL', 'CARD BILL\_JUN', 'CARD BILL\_MAY', 'CARD BILL\_APR']   
for i in range(len(CardBillVariable)):  
 Q1\_cb = dataframe[CardBillVariable[i]].quantile(0.25)  
 Q3\_cb = dataframe[CardBillVariable[i]].quantile(0.75)  
 IQR\_cb = Q3\_cb - Q1\_cb  
 dataframe\_CB = dataframe.loc[(dataframe[CardBillVariable[i]] > Q1\_cb - 0.5 \* IQR\_cb) & (dataframe[CardBillVariable[i]] < Q3\_cb + 0.5 \* IQR\_cb)]

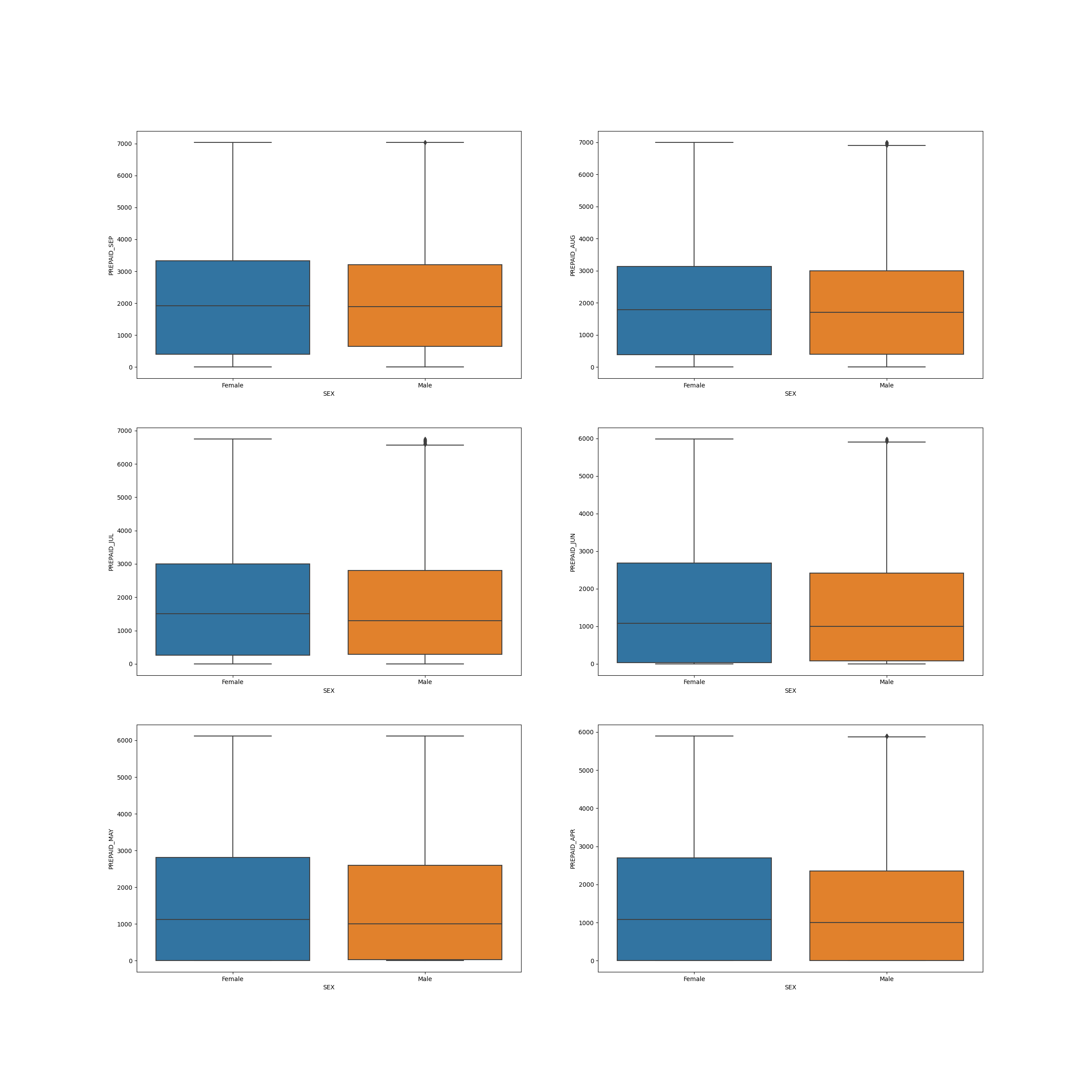
print(dataframe.groupby(['SEX'])[CardBillVariable[i]].median())

>>> Female 24110.0 Male 27284.0 Name: CARD BILL\_SEP, dtype: float64

Female 23201.0 Male 26015.5 Name: CARD BILL\_AUG, dtype: float64 Female 21865.0 Male 23610.0 Name: CARD BILL\_JUL, dtype: float64 Female 20204.0 Male 20245.0 Name: CARD BILL\_JUN, dtype: float64 Female 19225.0 Male 19374.0 Name: CARD BILL\_MAY, dtype: float64 Female 18375.0 Male 19066.5 Name: CARD BILL\_APR, dtype: float64

신용카드 청구액 변수도 신용한도 변수와 마찬가지로 관측값의 구간을 제한했다. 이상치를 포함했을 때 신용카드 청구액 중앙값은 남성이 대체로 높은 것으로 나타났다. 남성은 여성보다 평균적으로 최대 약 12%가량 더 많은 금액을 신용카드로 사용했다.



PrepaidVariable = ['PREPAID\_SEP', 'PREPAID\_AUG', 'PREPAID\_JUL', 'PREPAID\_JUN', 'PREPAID\_MAY', 'PREPAID\_APR']   
for i in range(len(PrepaidVariable)):  
 Q1\_pr = dataframe[PrepaidVariable[i]].quantile(0.25)  
 Q3\_pr = dataframe[PrepaidVariable[i]].quantile(0.75)  
 IQR\_pr = Q3\_pr - Q1\_pr  
 dataframe\_PR = dataframe.loc[(dataframe[PrepaidVariable[i]] > Q1\_pr - 0.5 \* IQR\_pr) & (dataframe[PrepaidVariable[i]] < Q3\_pr + 0.5 \* IQR\_pr)]

print(dataframe.groupby(['SEX'])[PrepaidVariable[i]].median())

>>> Female 2230.0 Male 2184.0 Name: PREPAID\_SEP, dtype: float64

Female 2107.0 Male 2021.0 Name: PREPAID\_AUG, dtype: float64

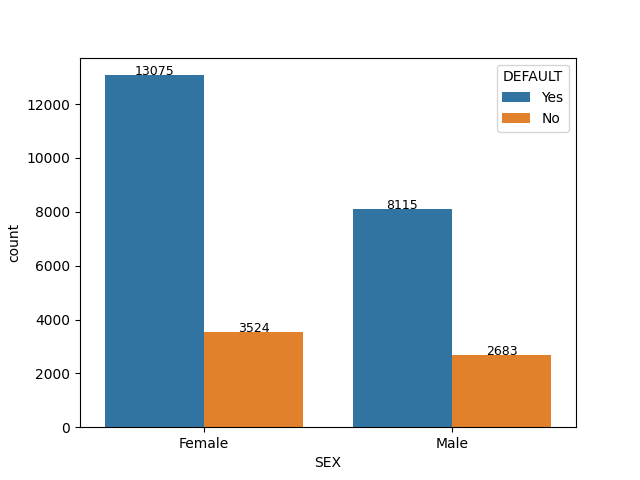
Female 2000.0 Male 1763.0 Name: PREPAID\_JUL, dtype: float64

Female 1650.0 Male 1500.0 Name: PREPAID\_JUN, dtype: float64

Female 1700.0 Male 1507.5 Name: PREPAID\_MAY, dtype: float64

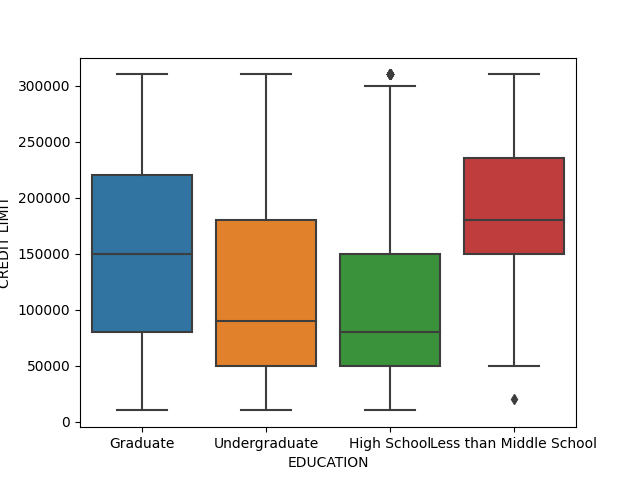
Female 1685.0 Male 1431.5 Name: PREPAID\_APR, dtype: float64

선불결제 이용액 변수도 앞의 경우와 같은 처리를 했다. 이상치 포함 중앙값은 남성보다 여성이 대체로 높았다. 여성은 평균적으로 남성보다 최대 17% 더 많은 금액을 선불결제로 사용했다.



채무 불이행 가능성, 즉 연체가 발생할 가능성은 그래프상에서 여성 약 21%, 남성 약 24%로 남성이 약간 더 높았다. 그러나 큰 의미를 부여할만한 차이는 아니다.

정리하면 부여받는 신용한도는 여성이 더 높았다. 신용카드 청구액 즉, 신용거래를 한 금액은 남성이 약 10%정도, 선불결제 이용액은 여성이 약 15% 정도 높았다. 채무불이행에 빠질 확률은 남성이 더 높았다. 신용카드사라면 신용거래를 더 많이하고 그만큼 연체할 확률이 높은 남성에게 상대적으로 약간 적은 한도를 부여하는 것이 합리적이라고 볼 수 있다.

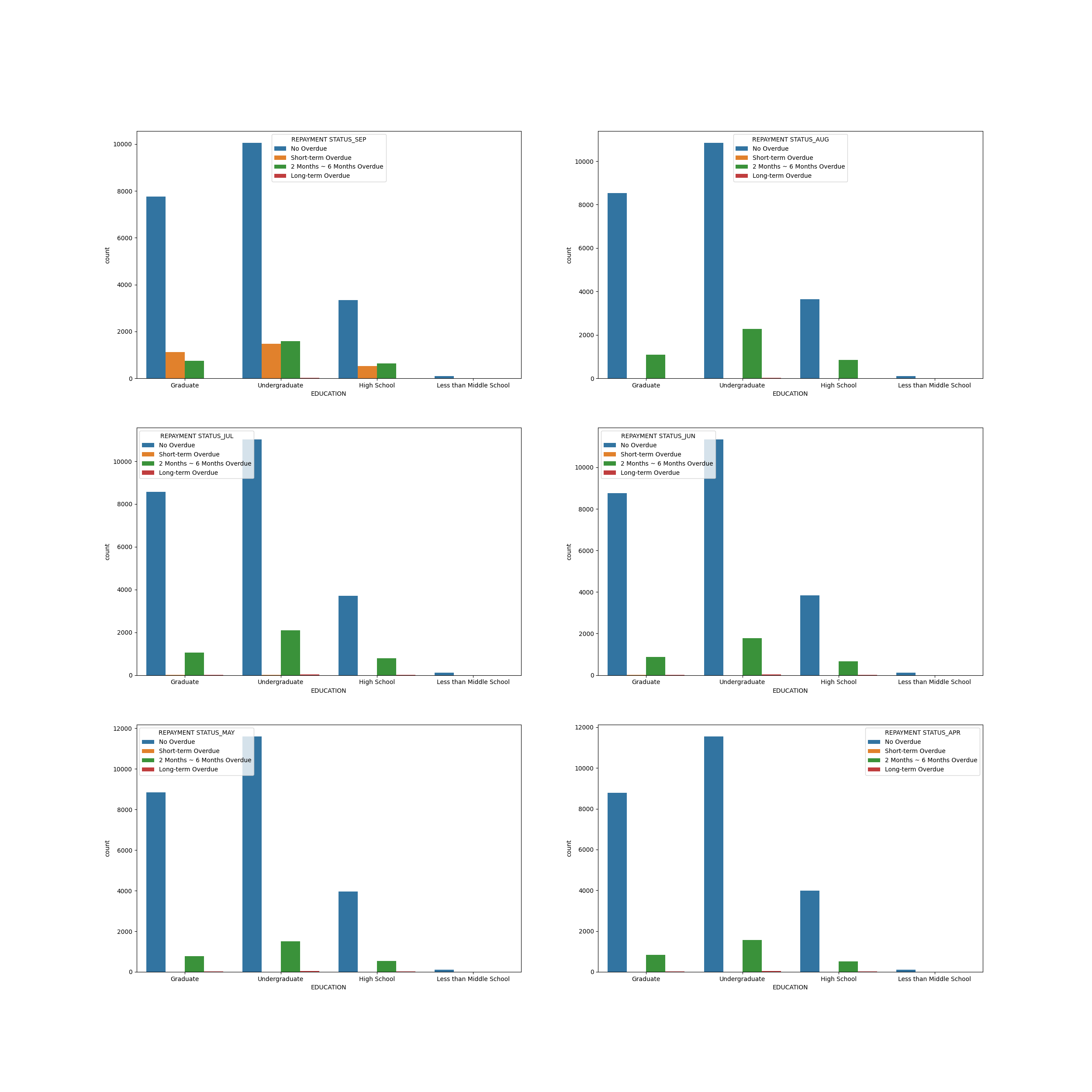
다음으로 학력 변수를 중심으로 데이터를 분석해보자.

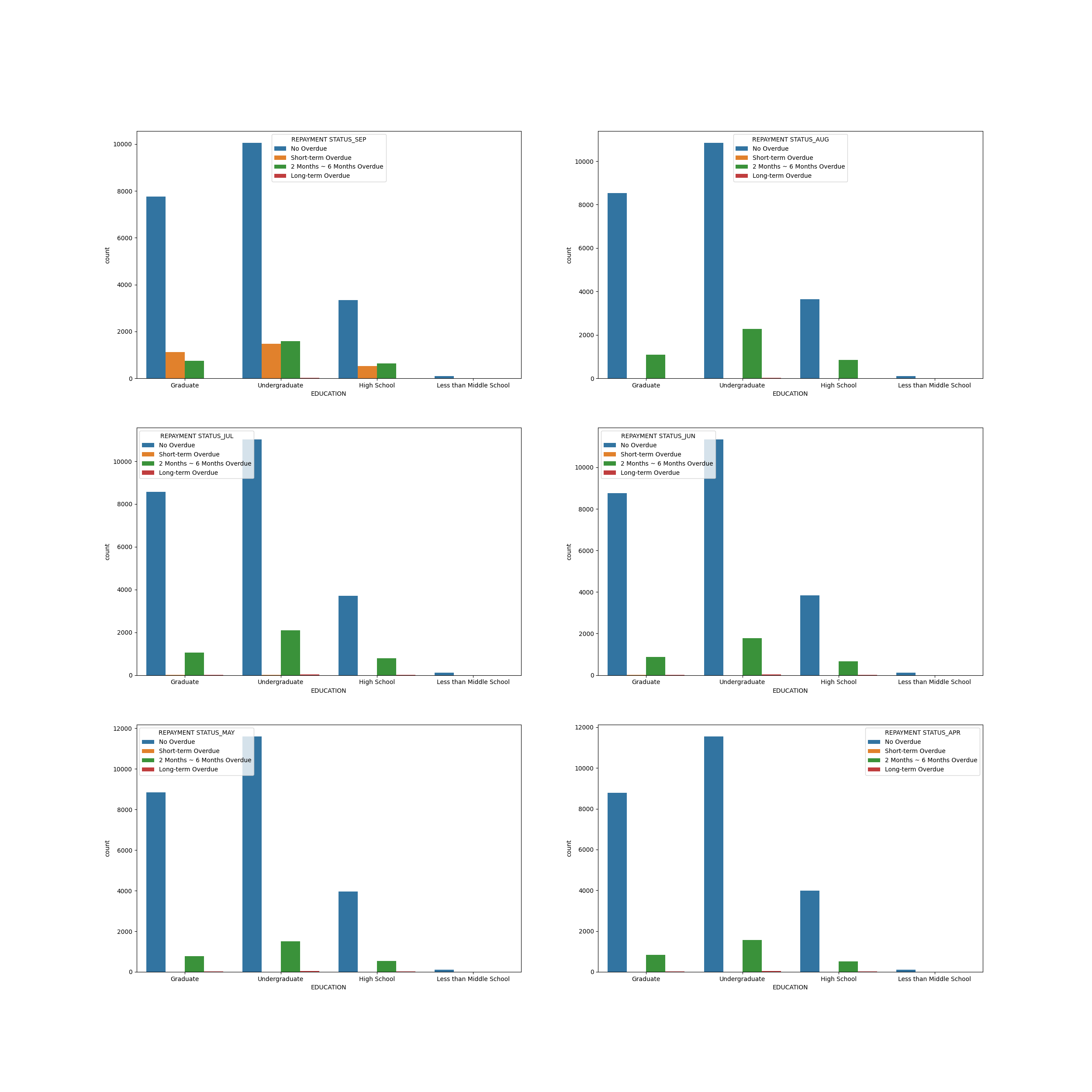
Q1\_cl = dataframe['CREDIT LIMIT'].quantile(0.25)  
Q3\_cl = dataframe['CREDIT LIMIT'].quantile(0.75)  
IQR\_cl = Q3\_cl - Q1\_cl  
dataframe\_CL = dataframe.loc[(dataframe['CREDIT LIMIT'] > Q1\_cl - 0.5 \* IQR\_cl) & (dataframe['CREDIT LIMIT'] < Q3\_cl + 0.5 \* IQR\_cl)]

print(dataframe.groupby(['EDUCATION'])['CREDIT LIMIT'].median())

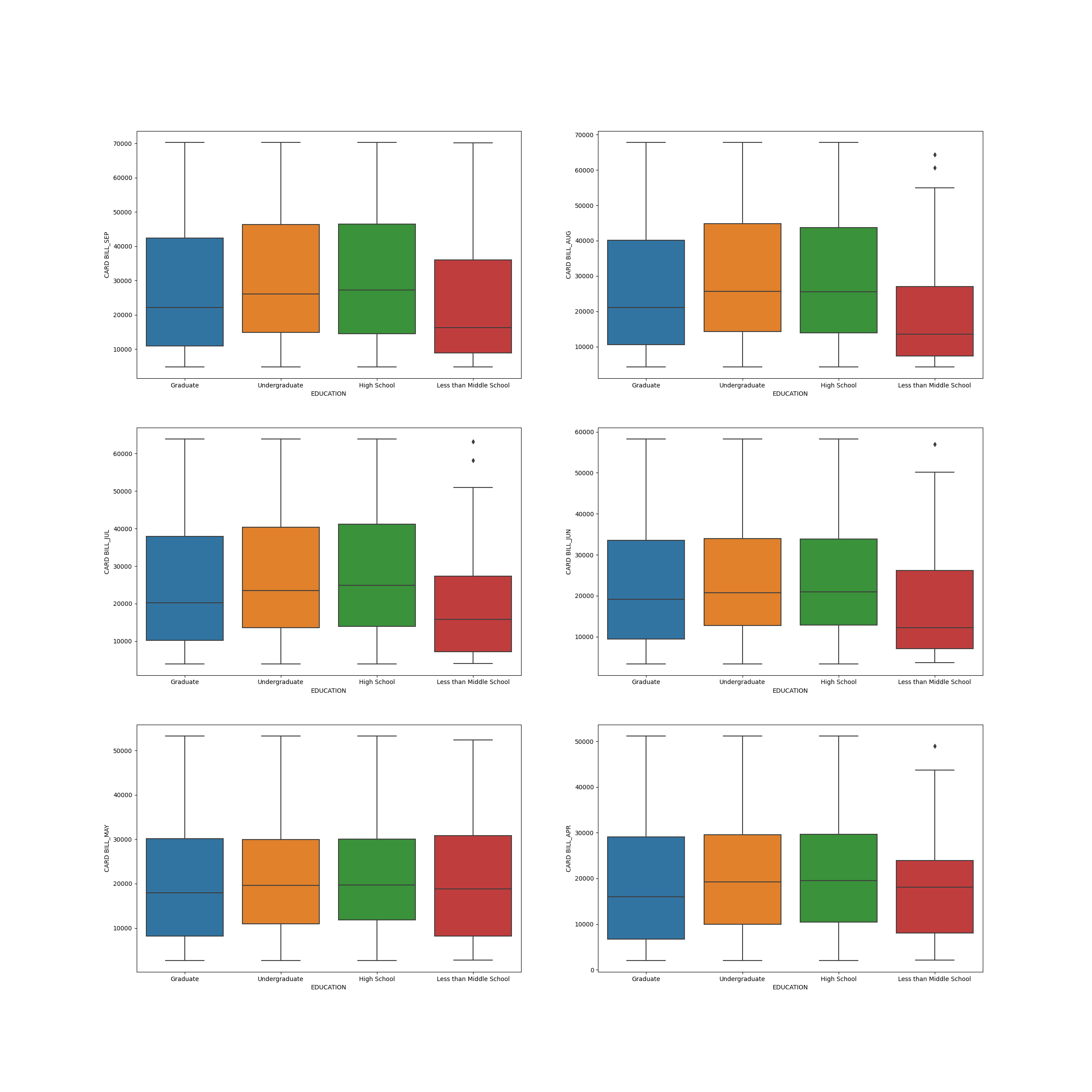
>>> Graduate 200000.0 Undergraduate 110000.0

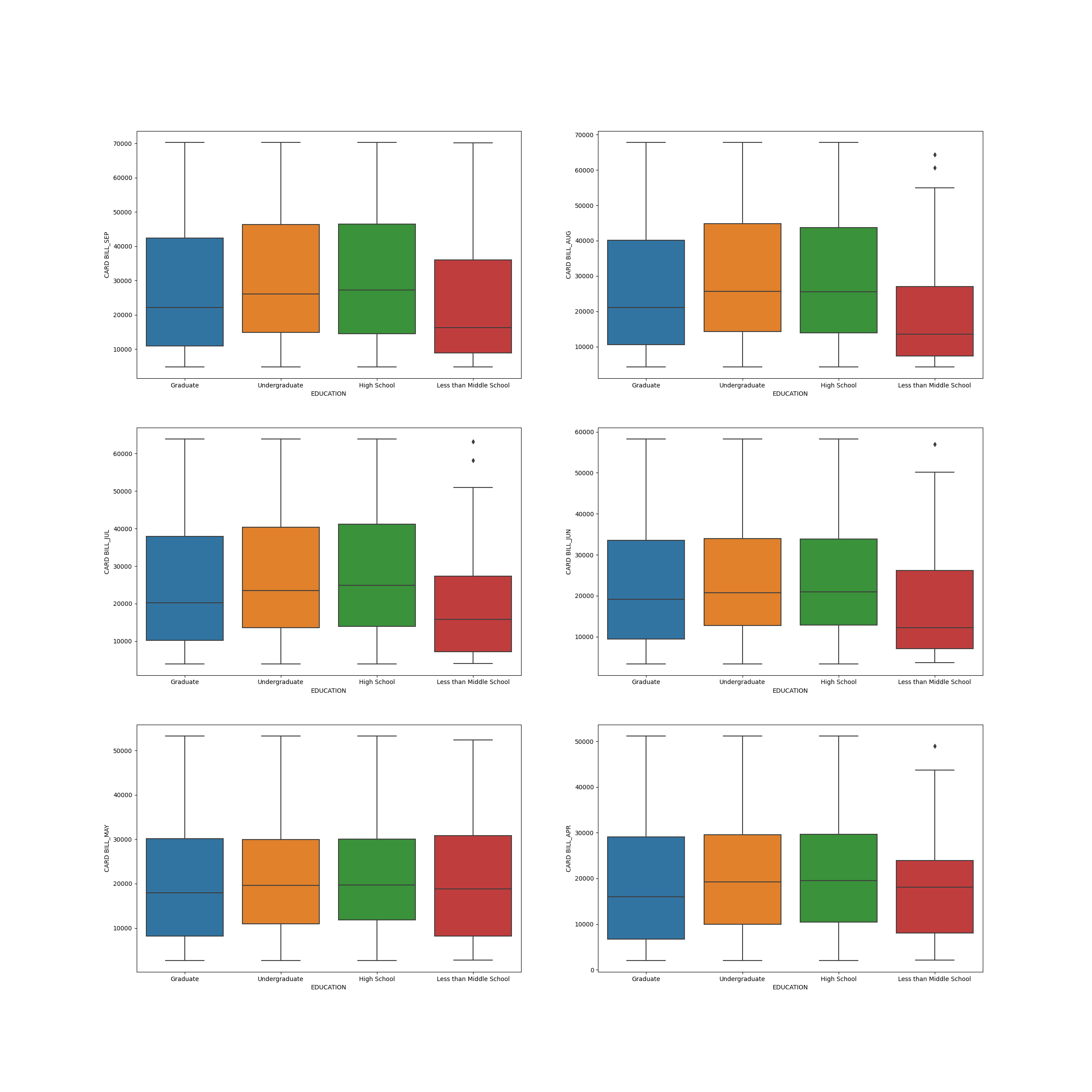
High School 80000.0 Less than Middle School 200000.0

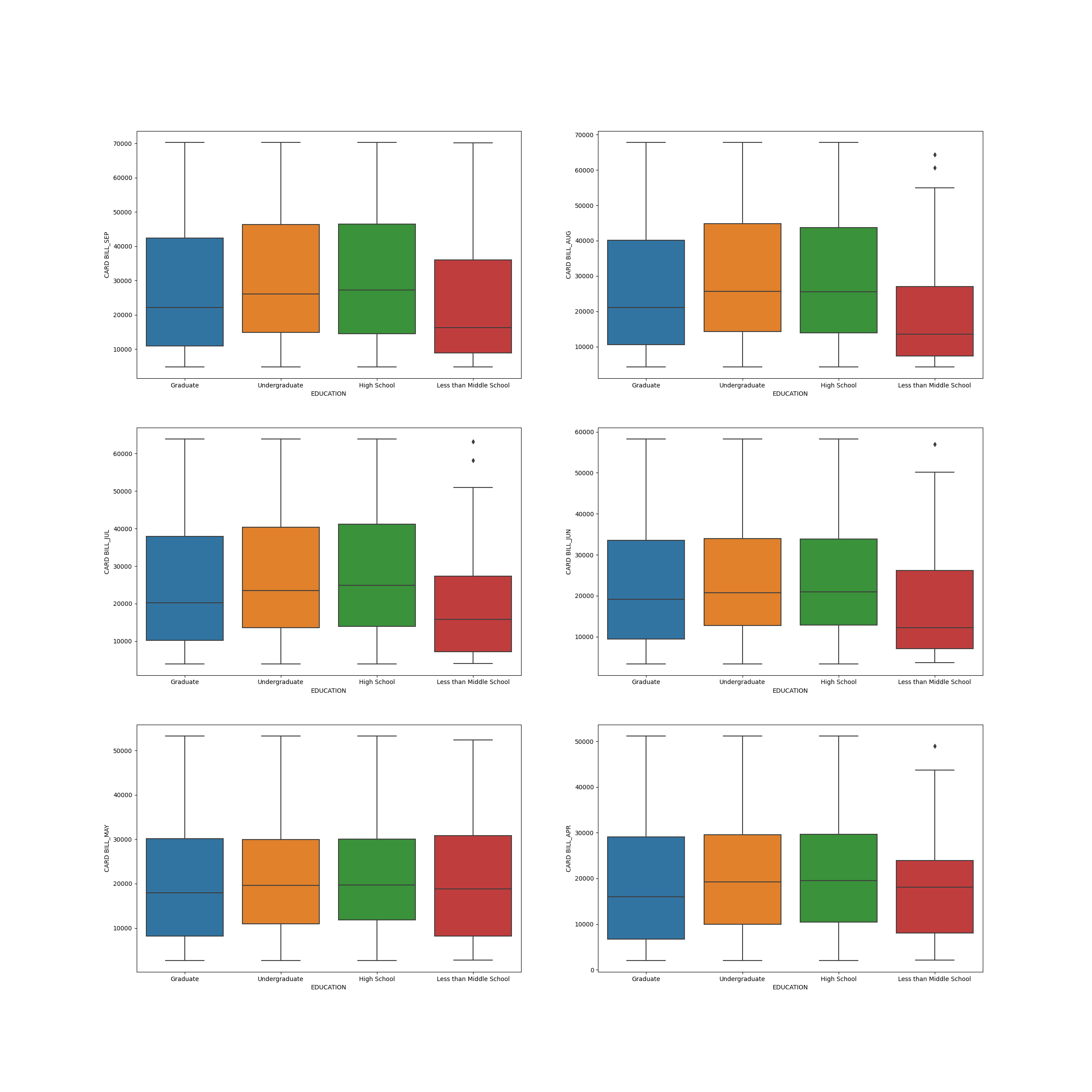
성별의 경우와 마찬가지로 변수에서 이상치가 보이지 않도록 관측값의 범위를 조절했다. 이때 표본이 적어 신뢰하기 어려운 중졸 이하의 표본을 제외하면 학력이 높아질수록 더 높은 신용한도를 부여받는 것으로 나타났다. 이상치를 포함할 때 평균적으로 대학생은 고졸자에 비해 37.5%, 대졸자는 고졸자에 비해 250% 높은 신용한도를 부여받는 것으로 나타났다. 고졸자와 대학생의 차이도 적지 않았지만 대졸자와 고졸자의 차이는 매우 컸다.



채무 상환 현황에 있어서는 학력이 낮을수록 상대적으로 연체의 비율이 높은 것으로 나타 났다. 다만 그 차이가 신용한도의 경우처럼 큰 차이는 아니었다. 대부분의 고객은 연체가 없었으며, 소수의 고객은 2개월에서 6개월의 연체가 있었다. 단기연체나 장기연체는 9월을 제외하고는 거의 없었다. 그래프 상에서 9월을 제외하고 대졸자 표본에서는 약 9% 정도의 고객이 연체가 있었고 대학생 및 고졸 표본에서는 약 11% ~ 17%의 고객이 연체가 있었다.







CardBillVariable = ['CARD BILL\_SEP', 'CARD BILL\_AUG', 'CARD BILL\_JUL', 'CARD BILL\_JUN', 'CARD BILL\_MAY', 'CARD BILL\_APR']   
for i in range(len(CardBillVariable)):  
 Q1\_cb = dataframe[CardBillVariable[i]].quantile(0.25)  
 Q3\_cb = dataframe[CardBillVariable[i]].quantile(0.75)  
 IQR\_cb = Q3\_cb - Q1\_cb  
 dataframe\_CB = dataframe.loc[(dataframe[CardBillVariable[i]] > Q1\_cb - 0 \* IQR\_cb) & (dataframe[CardBillVariable[i]] < Q3\_cb + 0 \* IQR\_cb)]

print(dataframe.groupby(['EDUCATION'])[CardBillVariable[i]].median())

>>> Graduate 17011.5 Graduate 17060.5

Undergraduate 29078.0 Undergraduate 28296.5

High School 27762.0 High School 26272.5

Less than Middle School 15019.0 Less than Middle School 11990.0

Name: CARD BILL\_SEP, dtype: float64 Name: CARD BILL\_AUG, dtype: float64 Graduate 17158.5 Graduate 16310.5

Undergraduate 26719.5 Undergraduate 23329.5

High School 25015.0 High School 20758.5

Less than Middle School 12681.0 Less than Middle School 11930.0

Name: CARD BILL\_JUL, dtype: float64 Name: CARD BILL\_JUN, dtype: float64

Graduate 14774.5 Graduate 12812.5

Undergraduate 20446.5 Undergraduate 20000.0

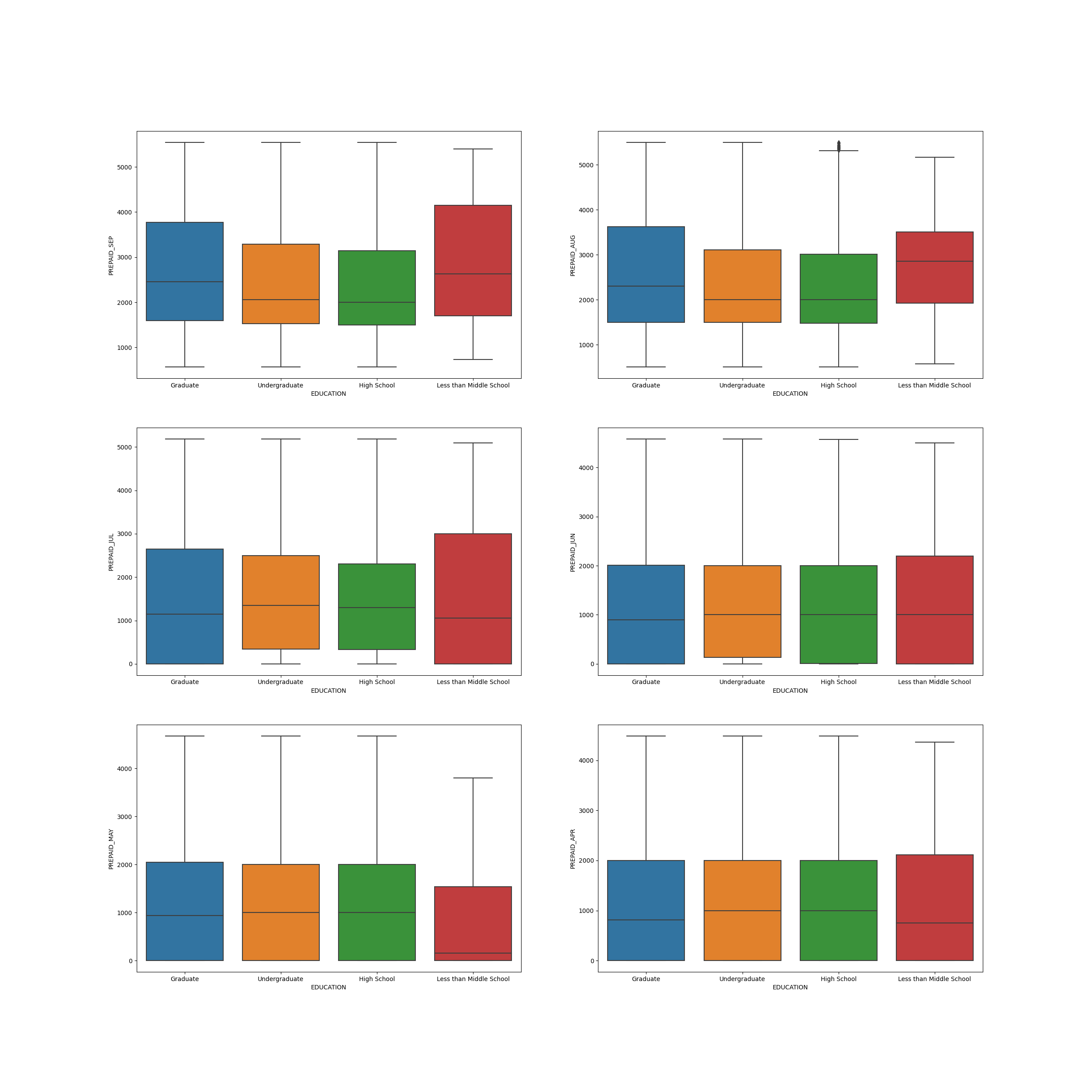
High School 19401.0 High School 19148.0

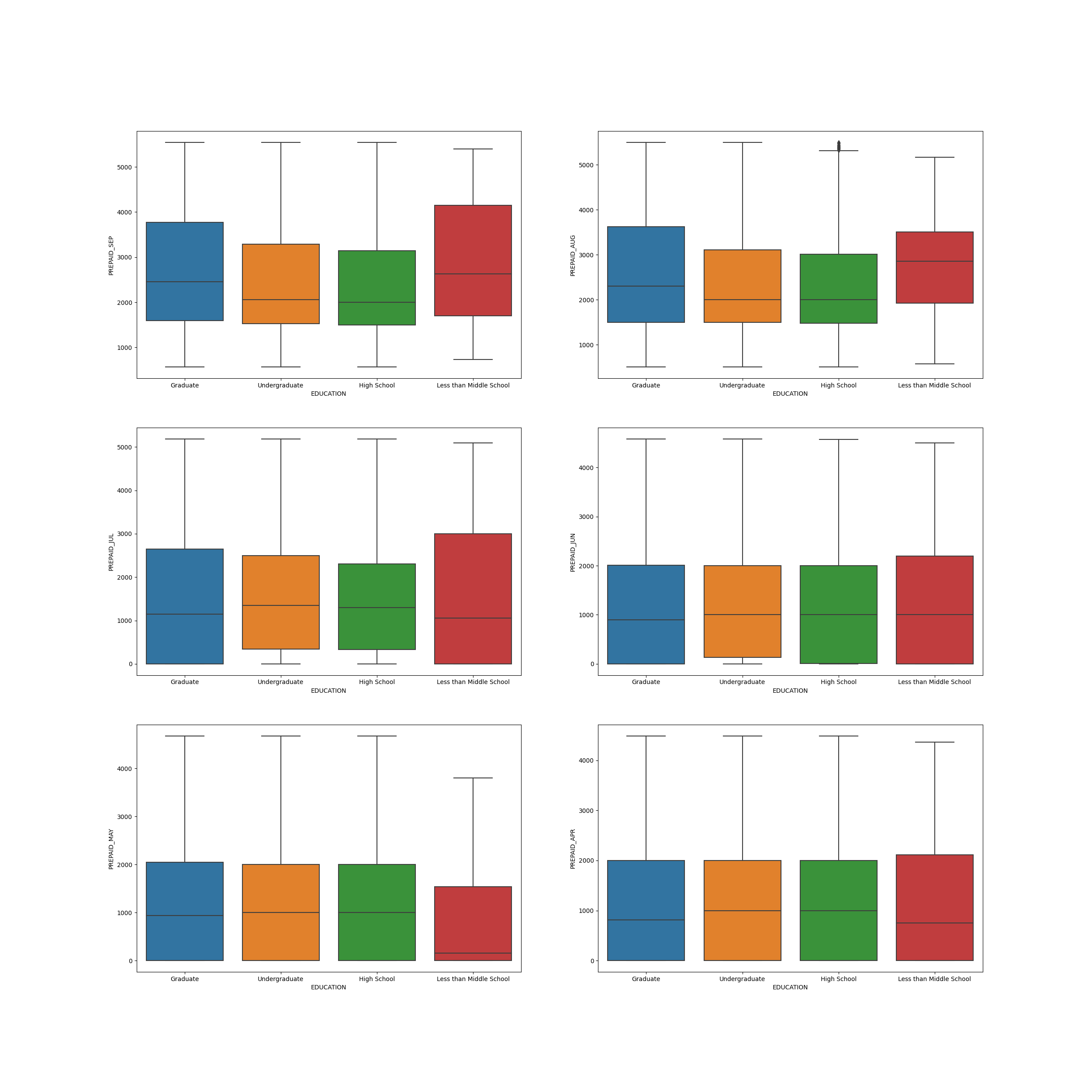
Less than Middle School 8539.0 Less than Middle School 8659.0

Name: CARD BILL\_MAY, dtype: float64 Name: CARD BILL\_APR, dtype: float64

신용카드 청구액에 있어서는 평균적으로 대졸자가 가장 낮았고 대학생과 고졸자가 대졸자

보다 높았다. 이상치를 포함할 때 대학생은 대졸자보다 신용카드 청구액이 최대 약 70%가량 높았으며, 고졸자는 대졸자보다 약 63%가량 높았다. 대학생과 고졸자는 평균적으로 신용카드로 결제한 금액이 대졸자보다 크게 높았지만 대학생과 고졸자는 차이가 크지 않았다. 중졸 이하는 표본이 적어 신뢰하기 힘들지만 신용카드 청구액이 가장 낮게 나왔다.





PrepaidVariable = ['PREPAID\_SEP', 'PREPAID\_AUG', 'PREPAID\_JUL', 'PREPAID\_JUN', 'PREPAID\_MAY', 'PREPAID\_APR']   
for i in range(len(PrepaidVariable)):  
 Q1\_pr = dataframe[PrepaidVariable[i]].quantile(0.25)  
 Q3\_pr = dataframe[PrepaidVariable[i]].quantile(0.75)  
 IQR\_pr = Q3\_pr - Q1\_pr  
 dataframe\_PR = dataframe.loc[(dataframe[PrepaidVariable[i]] > Q1\_pr - 0.125 \* IQR\_pr) & (dataframe[PrepaidVariable[i]] < Q3\_pr + 0.125 \* IQR\_pr)]

print(dataframe.groupby(['EDUCATION'])[PrepaidVariable[i]].median())

>>> Graduate 2700.0 Graduate 2557.0

Undergraduate 2100.0 Undergraduate 2002.0

High School 2000.0 High School 2000.0

Less than Middle School 3467.0 Less than Middle School 3016.0

Name: PREPAID\_SEP, dtype: float64 Name: PREPAID\_AUG, dtype: float64 Graduate 2000.0 Graduate 2000.0

Undergraduate 1831.0 Undergraduate 1523.0

High School 1601.5 High School 1258.5

Less than Middle School 3000.0 Less than Middle School 2110.0

Name: PREPAID\_JUL, dtype: float64 Name: PREPAID\_JUN, dtype: float64

Graduate 2000.0 Graduate 2000.0

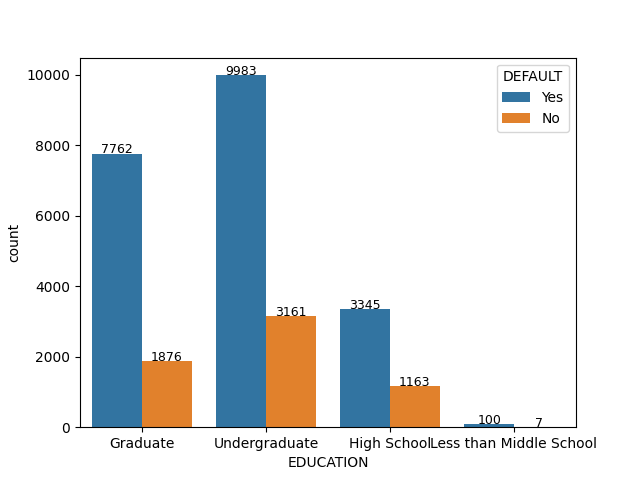
Undergraduate 1566.0 Undergraduate 1500.0

High School 1280.0 High School 1201.0

Less than Middle School 1523.0 Less than Middle School 2000.0

Name: PREPAID\_MAY, dtype: float64 Name: PREPAID\_APR, dtype: float64

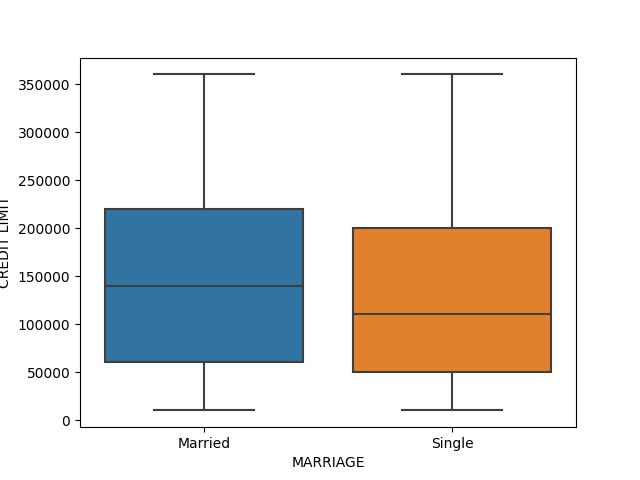
선불결제 이용액에 있어서는 대졸자가 가장 높았으며, 그다음이 대학생, 마지막으로 고졸자 수준이었다. 중졸 이하는 표본이 부족하므로 분석에서 제외한다. 이상치를 포함할 때 대졸자는 대학생보다 최대 33.3%가량 더 많이 선불결제했으며 고졸자보다는 최대 66.6%가량 더 많이 결제했다. 학력이 낮아짐에 따라 선불결제로 결제한 금액은 평균적으로 눈에 띄게 줄어들었다고 볼 수 있다.



채무 불이행 가능성, 즉 연체한 고객의 비율은 대졸자 약 19%, 대학생 약 24%, 고졸자 약 25%로 대졸자가 가장 낮았고 대학생과 고졸자는 비슷했다. 그래프상에서 대졸자는 연체할 가능성이 대학생이나 고졸자보다 절대적으로는 4~5%p, 상대적으로는 약 21%~24% 낮았다.

정리하면, 신용카드사는 대졸자에게 높은 신용한도를 부여하고 있고 대학생과 고졸자는 그보다 낮았다. 카드 사용 패턴에 있어서는 대학생과 고졸자가 대졸자보다 신용 카드 사용액이 눈에 띄게 높았고 선불결제 사용액은 눈에 띄게 낮았다. 또한 연체 가능성에 있어서도 대졸자보다 대학생, 고졸자가 더 높았다. 보통 대학 졸업 후 고정소득이 발생하는 것을 생각해보면 대학생, 고졸자는 소득대비 신용결제를 더 많이하는 것이므로 연체의 가능성또한 높은 것이다. 따라서 신용카드사라면 대졸자보다 대학생, 고졸자에게 신용한도를 매우 보수적으로 부여하는 것이 합리적이다.

그 다음으로는 결혼여부를 중심으로 데이터를 분석해보자.



Q1\_cl = dataframe['CREDIT LIMIT'].quantile(0.25)  
Q3\_cl = dataframe['CREDIT LIMIT'].quantile(0.75)  
IQR\_cl = Q3\_cl - Q1\_cl  
dataframe\_CL = dataframe.loc[(dataframe['CREDIT LIMIT'] > Q1\_cl - 0.75 \* IQR\_cl) & (dataframe['CREDIT LIMIT'] < Q3\_cl + 0.75 \* IQR\_cl)]

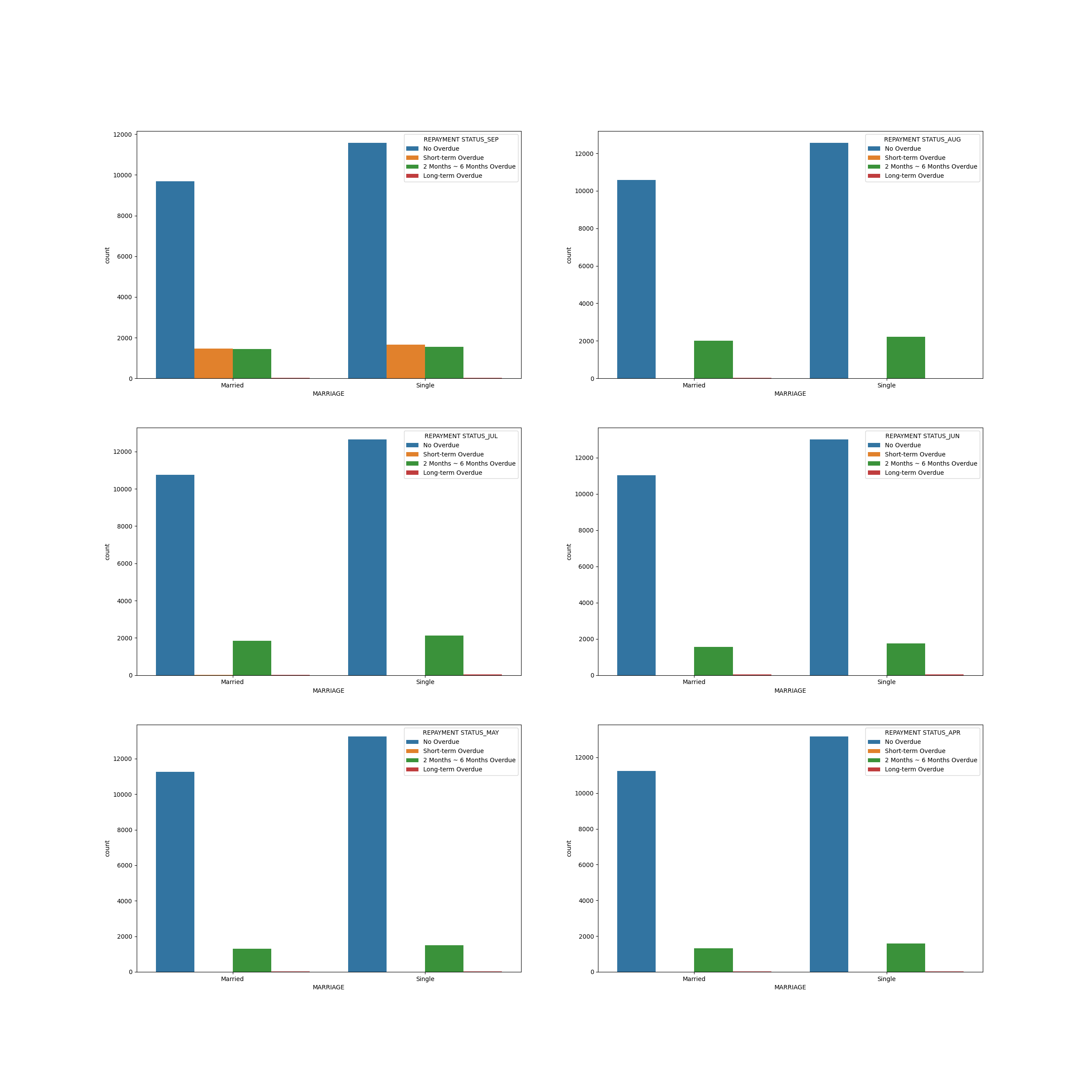
print(dataframe.groupby(['MARRIAGE'])['CREDIT LIMIT'].median())

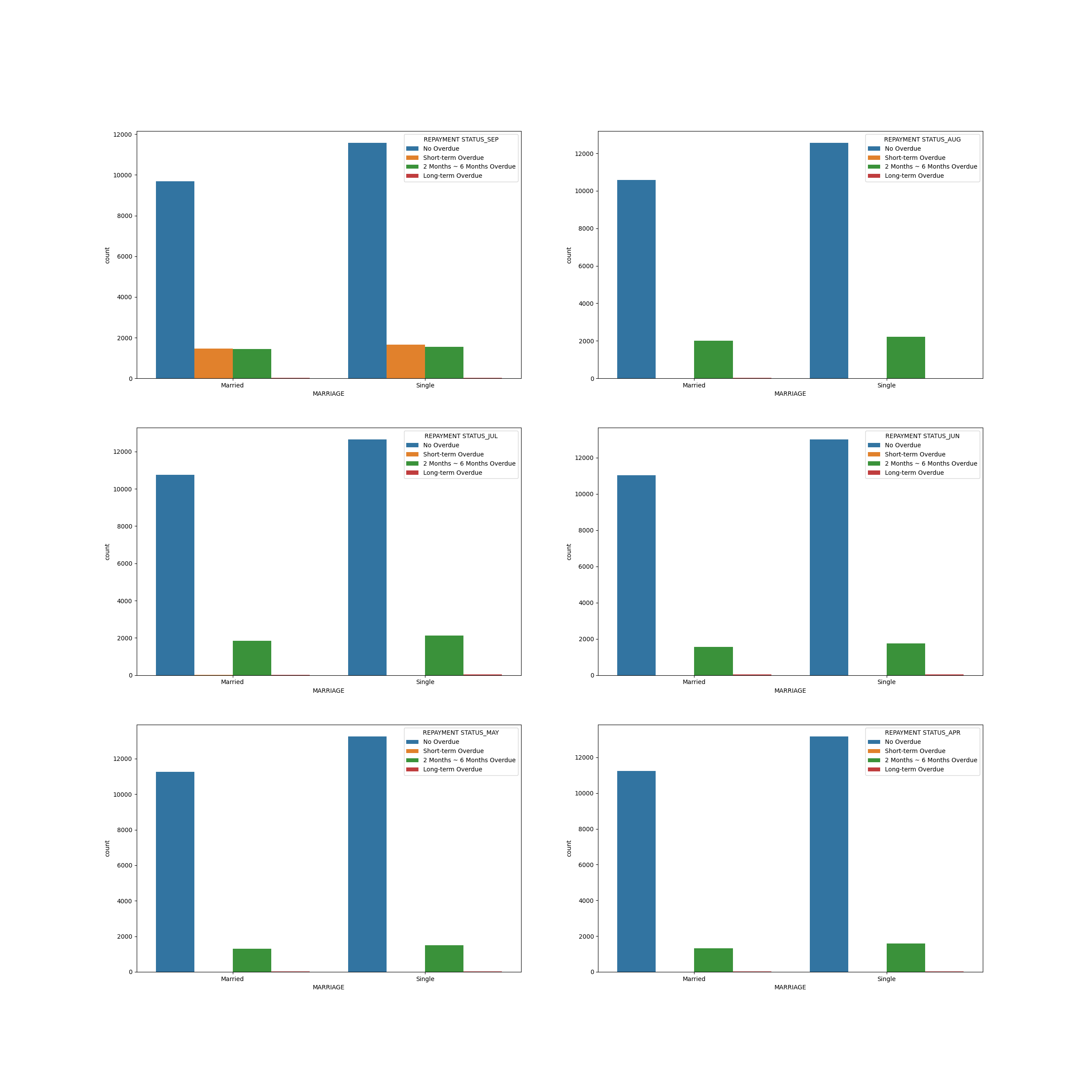
>>> Married 150000.0

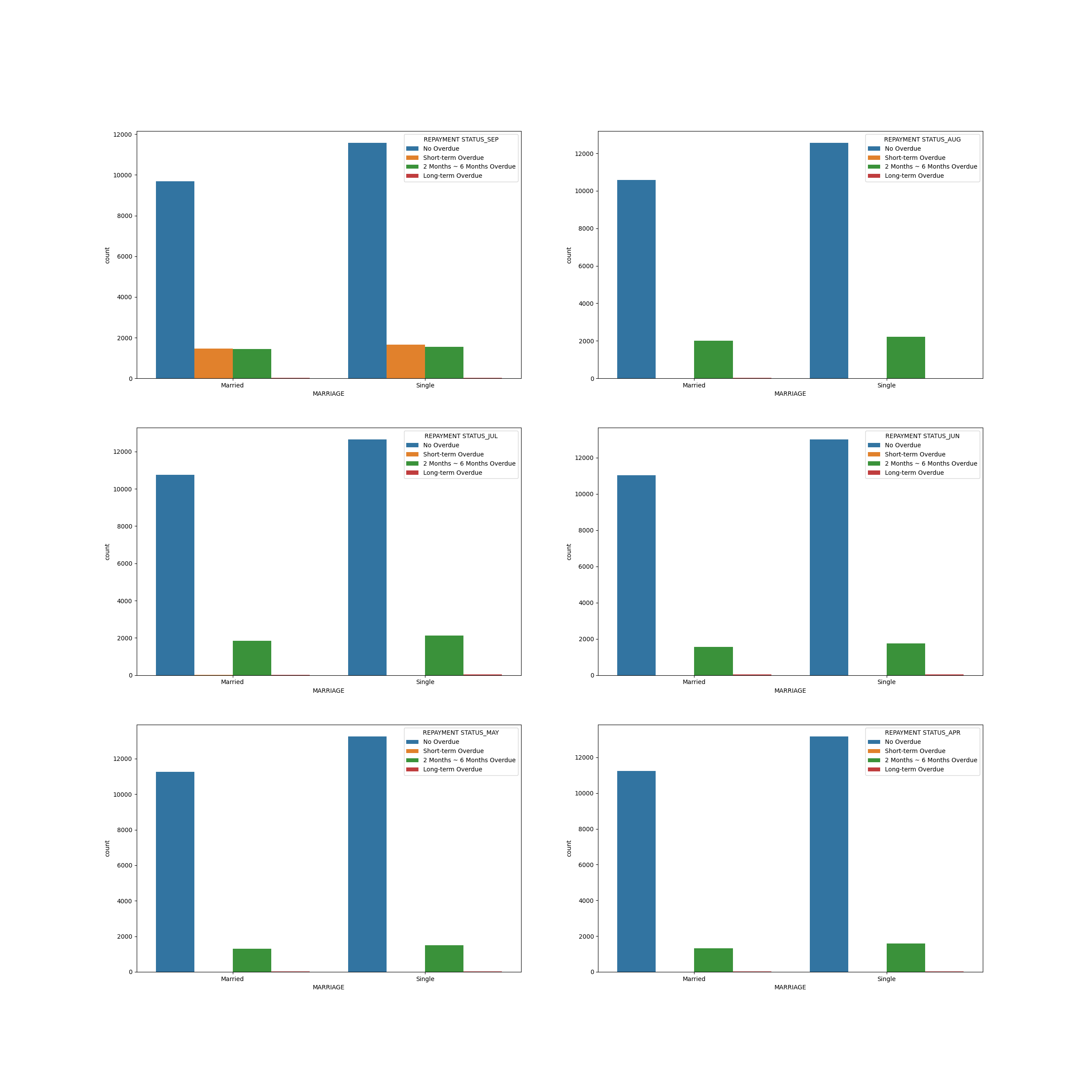
Single 120000.0

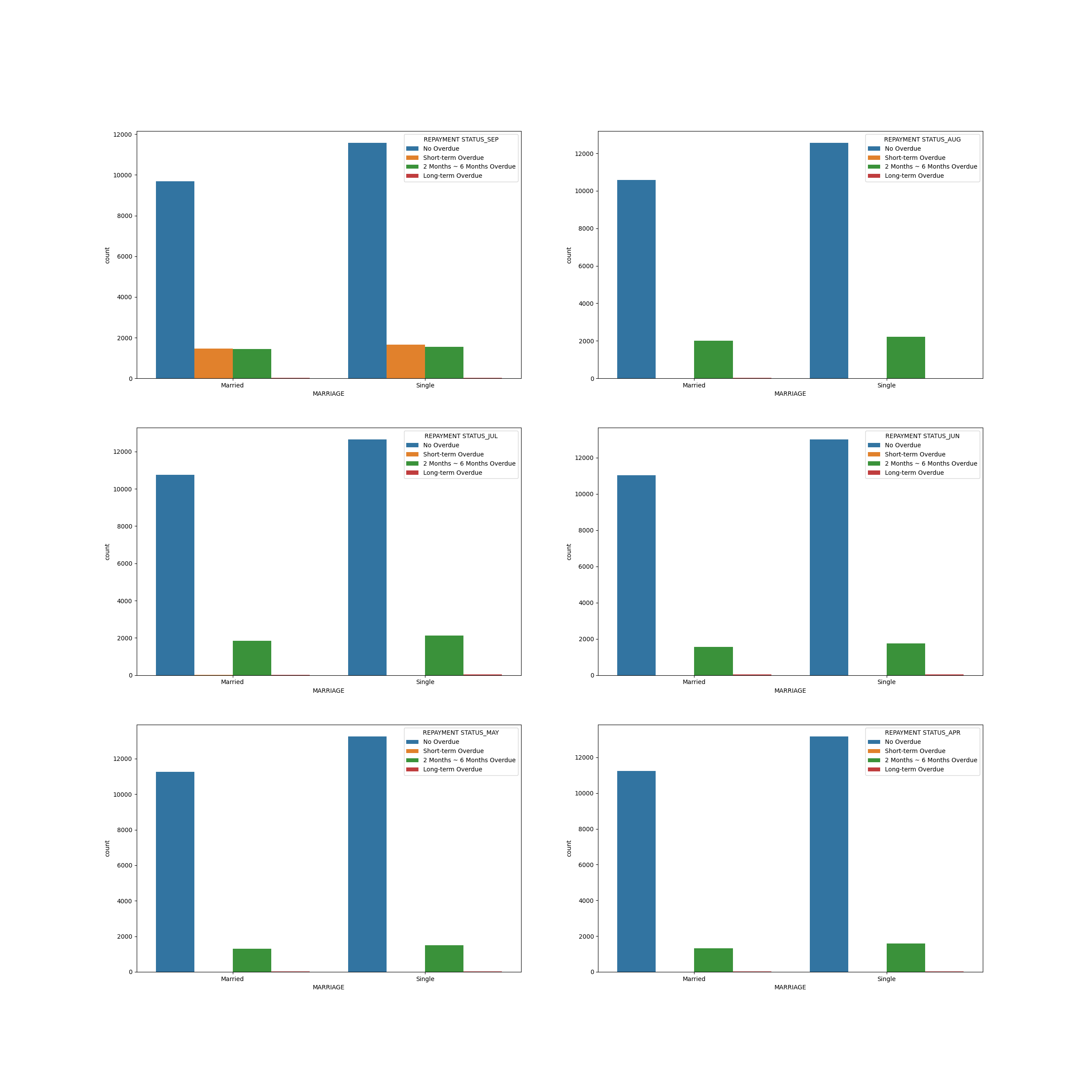
Name: CREDIT LIMIT, dtype: float64

다른 변수에서와 마찬가지로 이상치가 그래프에 나타나지 않도록 관측값의 범위를 조절했다. 이때 신용한도는 기혼자가 미혼보다 더 높았다. 평균적으로 기혼자가 미혼보다 신용한도를 25%가량 높게 부여받는 것으로 나타났다.

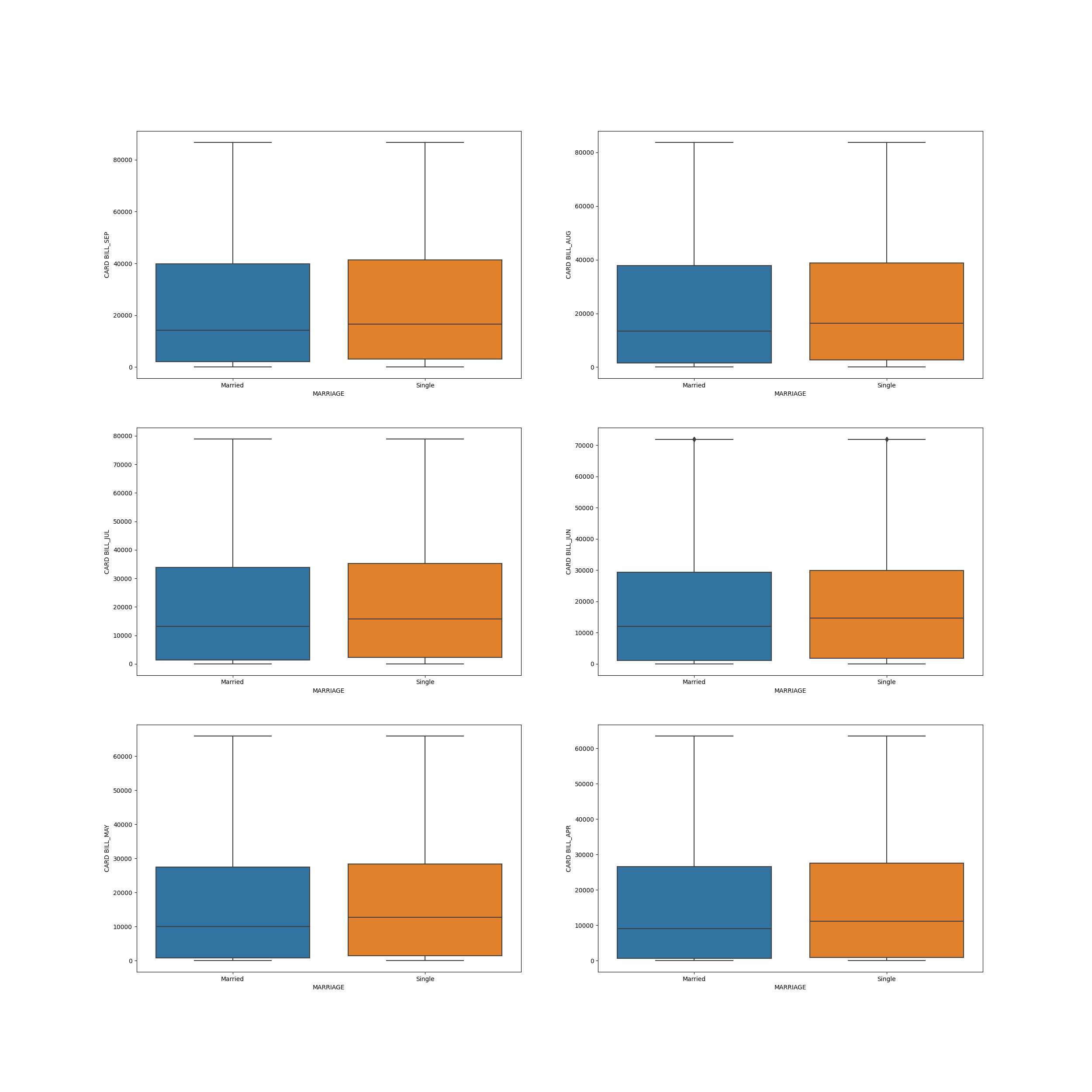


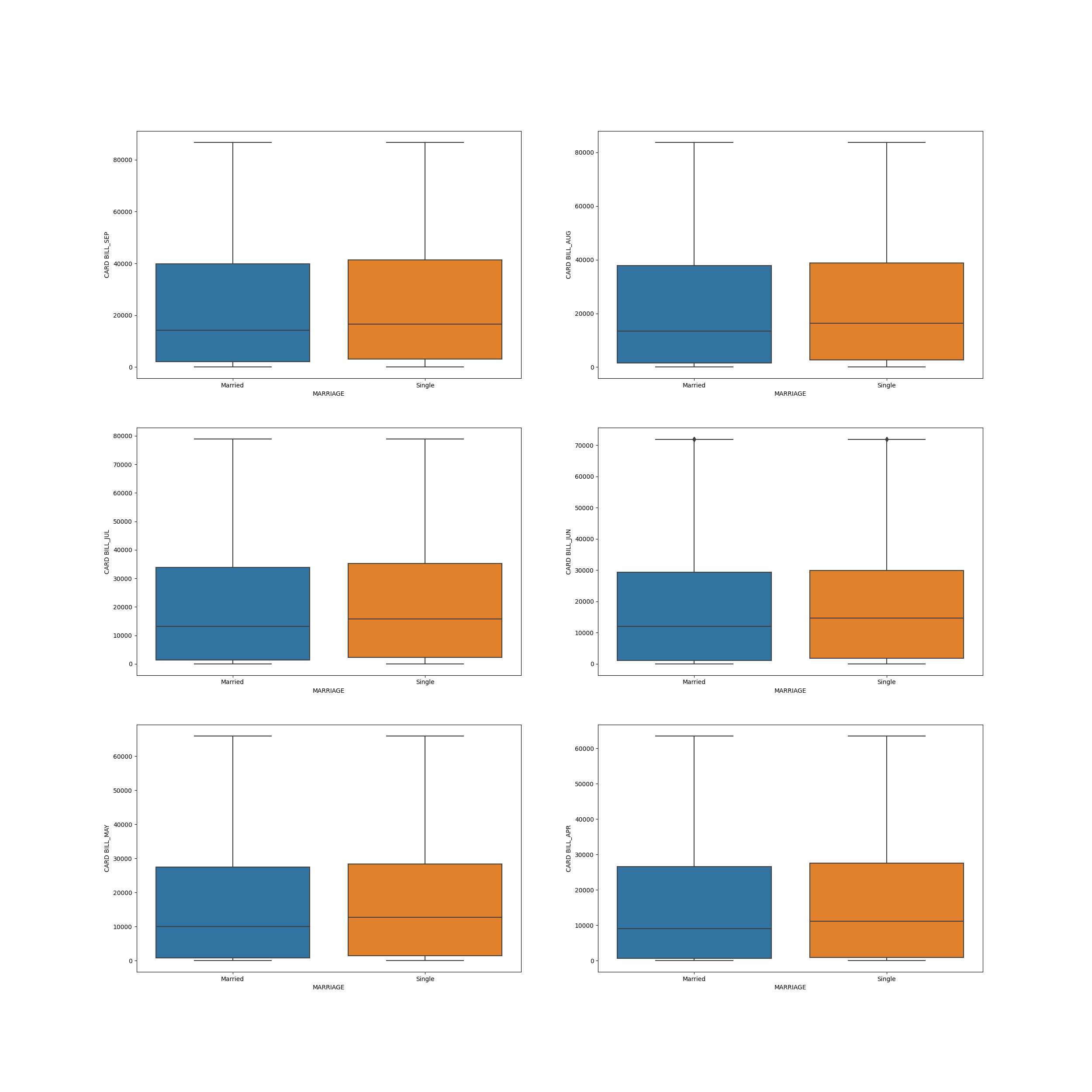


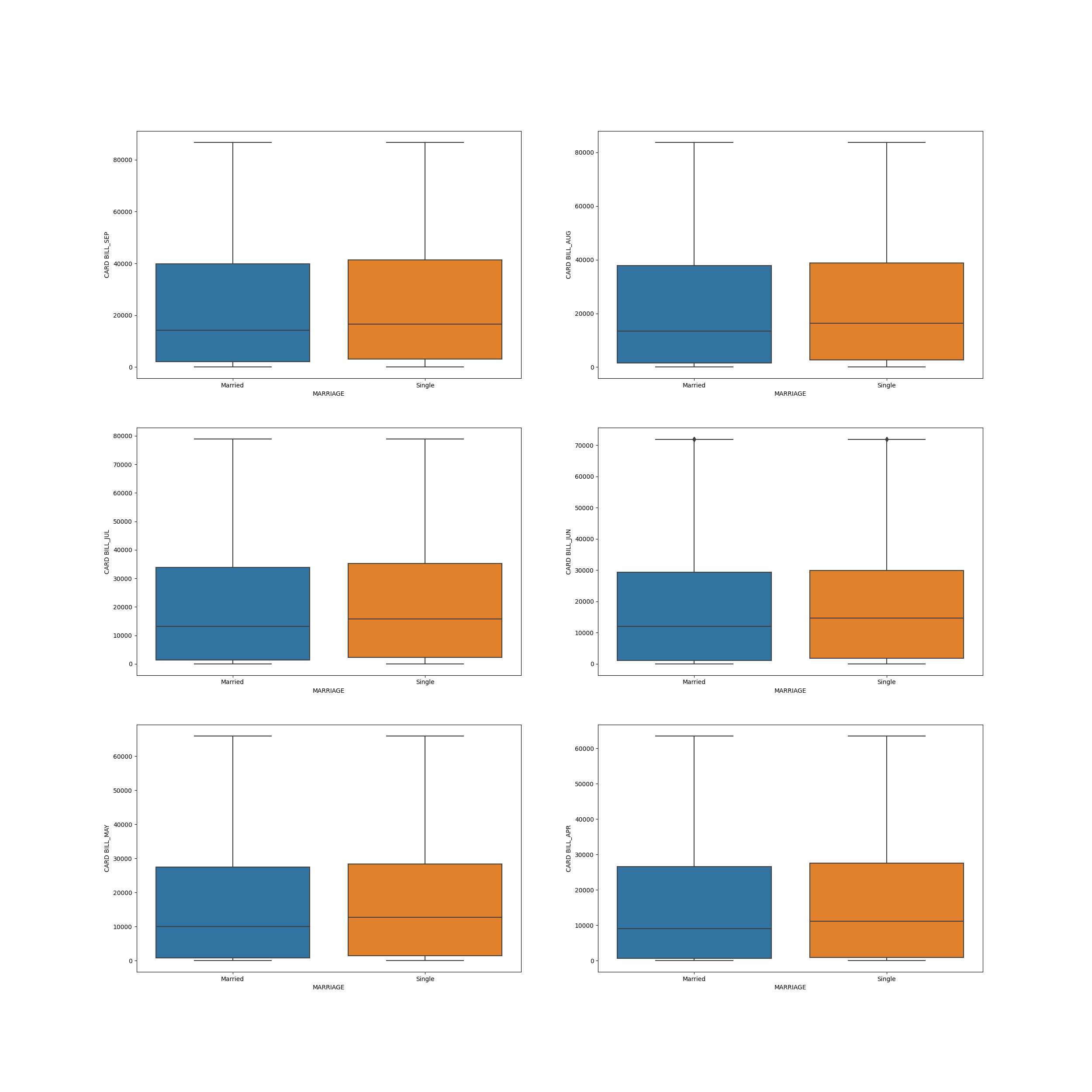




카드 대금 상환 현황에 있어서는 결혼여부에 따라 차이가 드러나지 않았다. 연체가 없는 사람이 대다수였고, 연체가 있는 경우에도 2개월에서 6개월정도의 연체가 대부분이었다. 그래프상에서 확인해보면 결혼여부에 상관없이 약 13%~14%정도의 고객이 연체가 있었다.







CardBillVariable = ['CARD BILL\_SEP', 'CARD BILL\_AUG', 'CARD BILL\_JUL', 'CARD BILL\_JUN', 'CARD BILL\_MAY', 'CARD BILL\_APR']   
for i in range(len(CardBillVariable)):  
 Q1\_cb = dataframe[CardBillVariable[i]].quantile(0.25)  
 Q3\_cb = dataframe[CardBillVariable[i]].quantile(0.75)  
 IQR\_cb = Q3\_cb - Q1\_cb  
 dataframe\_CB = dataframe.loc[(dataframe[CardBillVariable[i]] > Q1\_cb - 0.25 \* IQR\_cb) & (dataframe[CardBillVariable[i]] < Q3\_cb + 0.25 \* IQR\_cb)]

print(dataframe.groupby(['MARRIAGE'])[CardBillVariable[i]].median())

>>> Married 25395.0 Single 25263.0 Name: CARD BILL\_SEP, dtype: float64

Married 23978.0 Single 24583.0 Name: CARD BILL\_AUG, dtype: float64

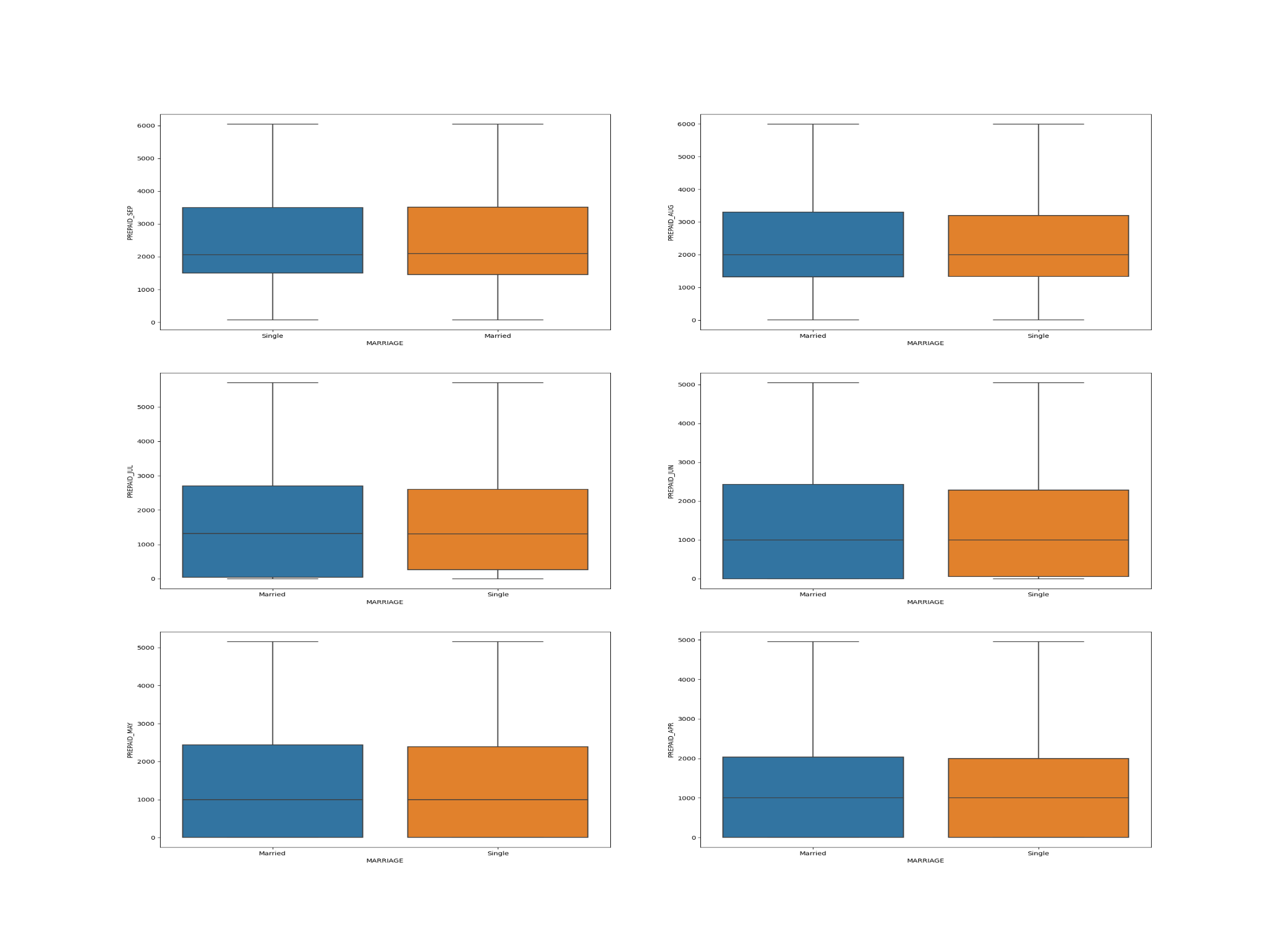
Married 22456.0 Single 22719.5 Name: CARD BILL\_JUL, dtype: float64

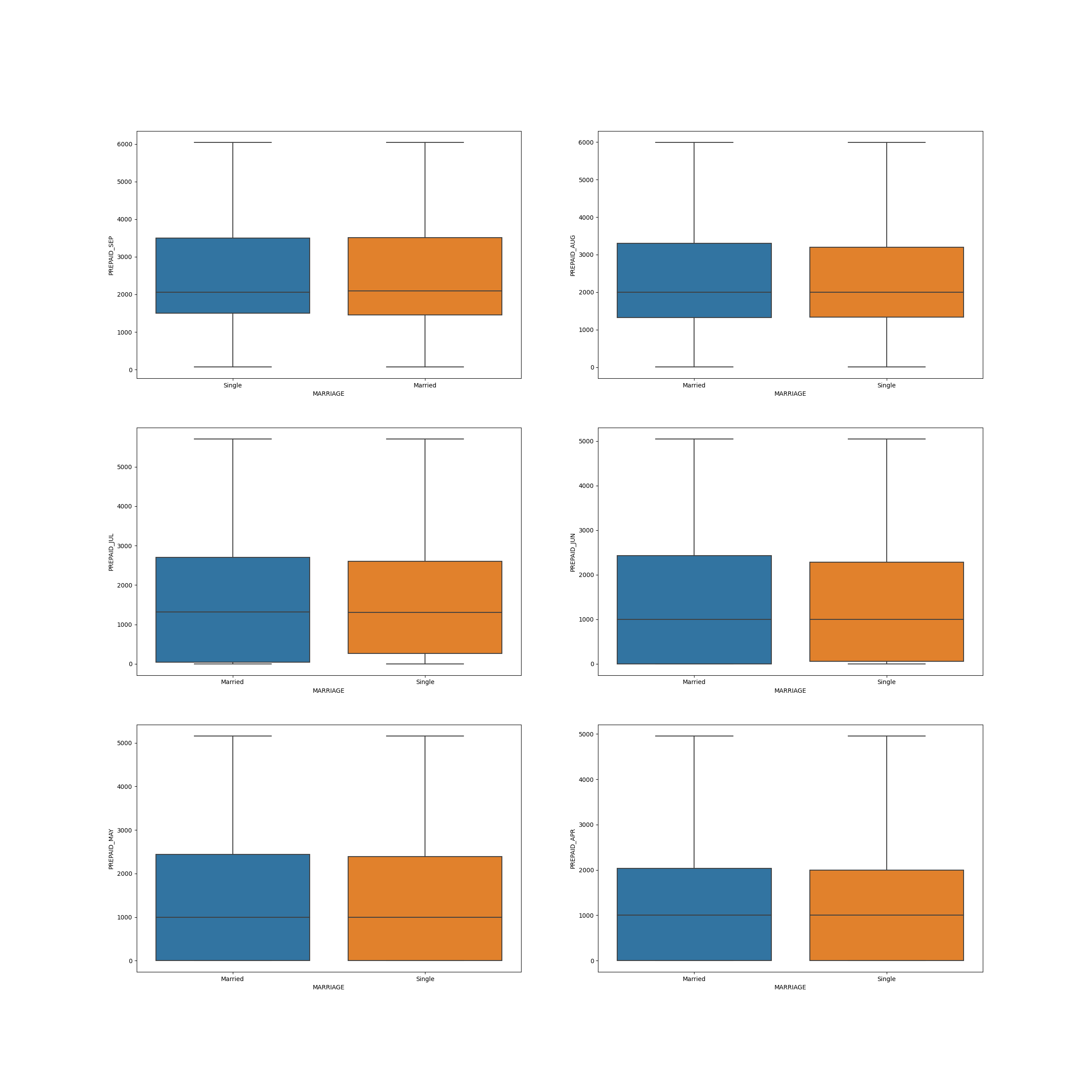
Married 20204.0 Single 20245.0 Name: CARD BILL\_JUN, dtype: float64

Married 19221.0 Single 19372.0 Name: CARD BILL\_MAY, dtype: float64

Married 18455.0 Single 18976.0 Name: CARD BILL\_APR, dtype: float64

신용카드 청구액에 있어서는 결혼여부에 따라 차이가 없었다. 이상치를 포함했을 때 평균적으로 미혼자가 조금 더 높았으나 큰 차이는 아니었고 기혼자가 더 높은 시기도 있었다. 차이또한 최대 2% 정도의 차이에 불과했다.





PrepaidVariable = ['PREPAID\_SEP', 'PREPAID\_AUG', 'PREPAID\_JUL', 'PREPAID\_JUN', 'PREPAID\_MAY', 'PREPAID\_APR']   
for i in range(len(PrepaidVariable)):  
 Q1\_pr = dataframe[PrepaidVariable[i]].quantile(0.25)  
 Q3\_pr = dataframe[PrepaidVariable[i]].quantile(0.75)  
 IQR\_pr = Q3\_pr - Q1\_pr  
 dataframe\_PR = dataframe.loc[(dataframe[PrepaidVariable[i]] > Q1\_pr - 0.25 \* IQR\_pr) & (dataframe[PrepaidVariable[i]] < Q3\_pr + 0.25 \* IQR\_pr)]

print(dataframe.groupby(['MARRIAGE'])[PrepaidVariable[i]].median())

>>> Married 2228.0 Single 2200.0 Name: PREPAID\_SEP, dtype: float64

Married 2123.0 Single 2036.0 Name: PREPAID\_AUG, dtype: float64

Married 2000.0 Single 1889.0 Name: PREPAID\_JUL, dtype: float64

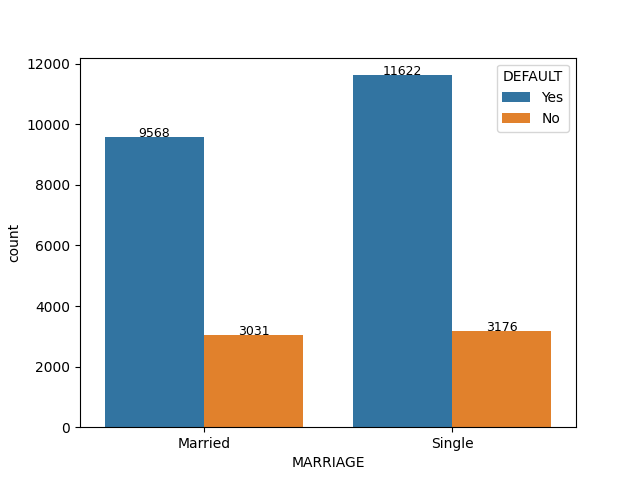
Married 1642.0 Single 1503.0 Name: PREPAID\_JUN, dtype: float64

Married 1666.0 Single 1600.0 Name: PREPAID\_MAY, dtype: float64

Married 1588.0 Single 1500.0 Name: PREPAID\_APR, dtype: float64

선불결제 이용액 역시 결혼여부에 상관없이 유사한 수준이었다. 기혼자가 미혼자보다 선불

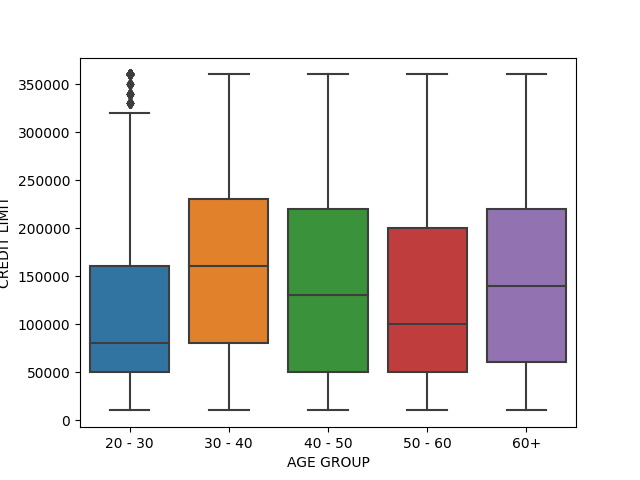
로 결제한 금액이 미세하게 더 높았지만 이상치를 포함해도 그 차이는 최대 9% 수준이었다.



채무 불이행 가능성, 즉 연체자 비율은 기혼자 약 24%, 미혼 약 21%로 그래프상에서 기혼자가 절대치로 3%p, 상대적으로 약 14% 높았다.

정리하면 신용카드사에서는 기혼자에게 미혼보다 더 높은 한도를 부여하고 있다. 그러나 사용 패턴을 살펴보면 신용카드로든 선불로든 평균적인 결제금액은 차이가 거의 없었고 연체 가능성만 기혼자가 소폭 높았다. 이때 기혼자와 미혼의 신용한도차이는 채무불이행 가능성의 차이보다 컸다. 따라서 신용카드사라면 미혼자에게 기혼자와 동일하거나 조금 낮은 한도를 부여하는 것이 더 좋은 선택일 수 있다. 미혼 고객에게 적절한 한도를 부여함으로서 더 많은 고객을 끌어들일 수 있으면서도 리스크는 최소화할 수 있다.

그럼 마지막으로 연령대를 중심으로 데이터를 분석해보자.



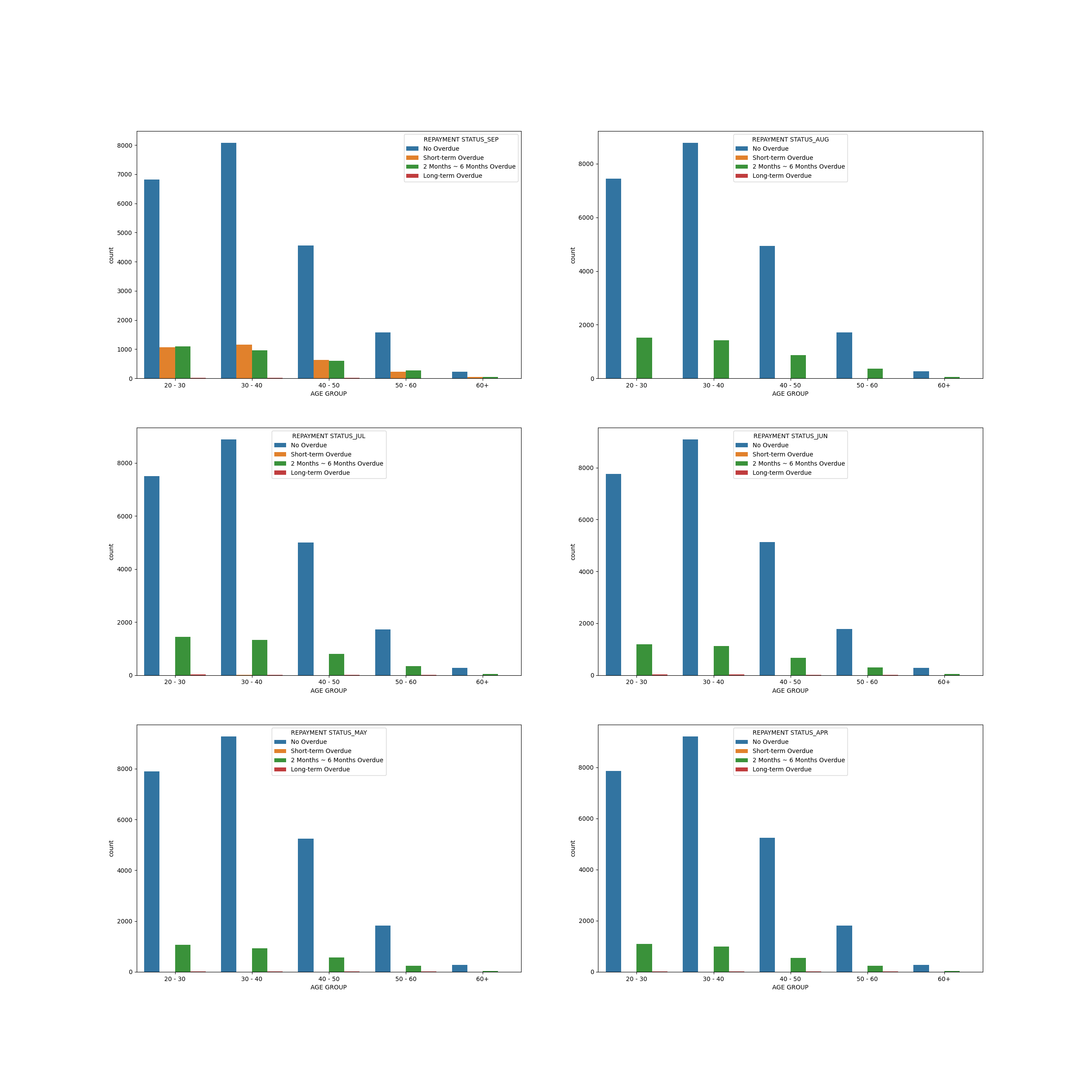
Q1\_cl = dataframe['CREDIT LIMIT'].quantile(0.25)  
Q3\_cl = dataframe['CREDIT LIMIT'].quantile(0.75)  
IQR\_cl = Q3\_cl - Q1\_cl  
dataframe\_CL = dataframe.loc[(dataframe['CREDIT LIMIT'] > Q1\_cl - 0.75 \* IQR\_cl) & (dataframe['CREDIT LIMIT'] < Q3\_cl + 0.75 \* IQR\_cl)]

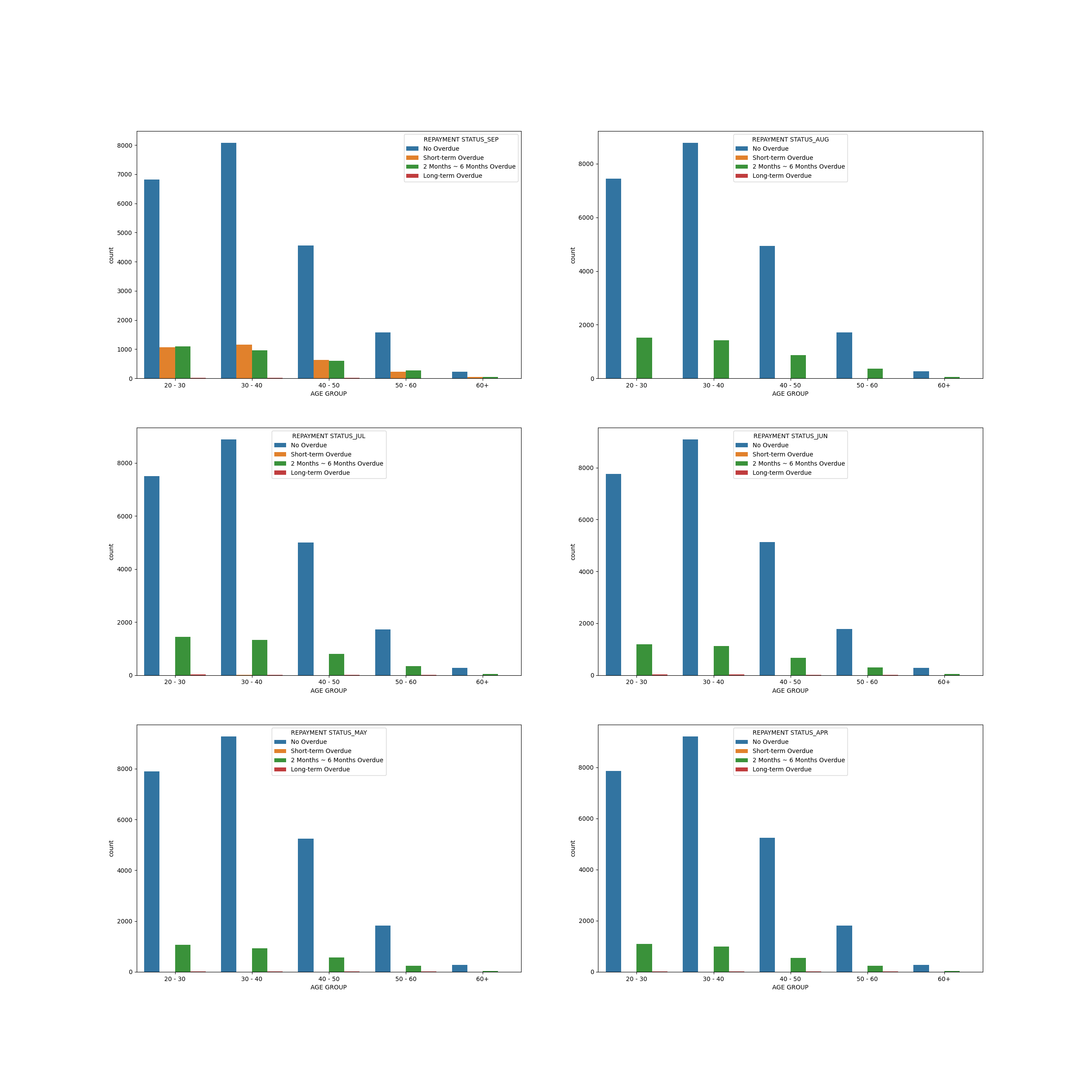
print(dataframe.groupby(['AGE GROUP'])['CREDIT LIMIT'].median())

>>> 20 - 30 90000.0 30 - 40 180000.0 40 - 50 150000.0

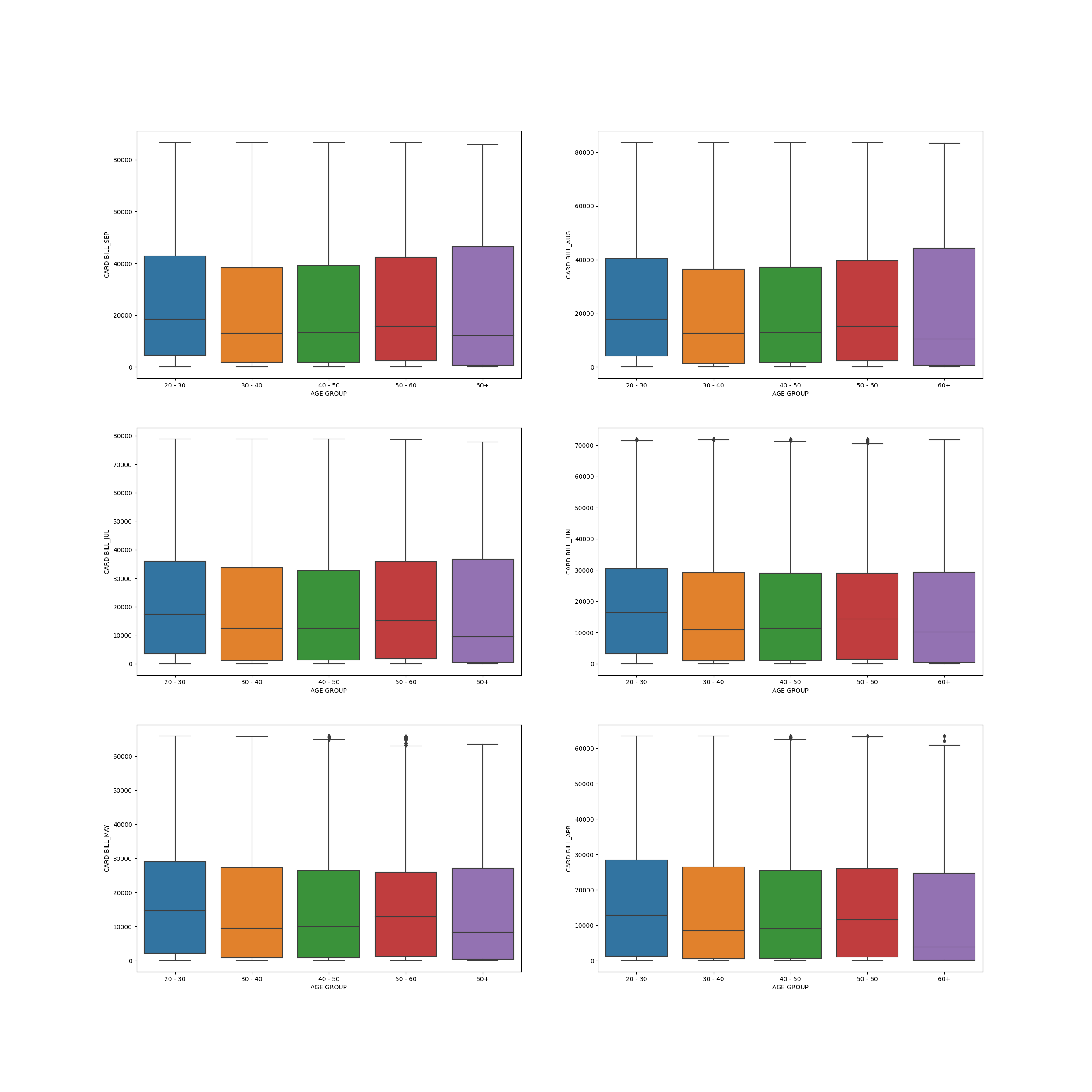
50 - 60 110000.0 60+ 160000.0 Name: CREDIT LIMIT, dtype: float64

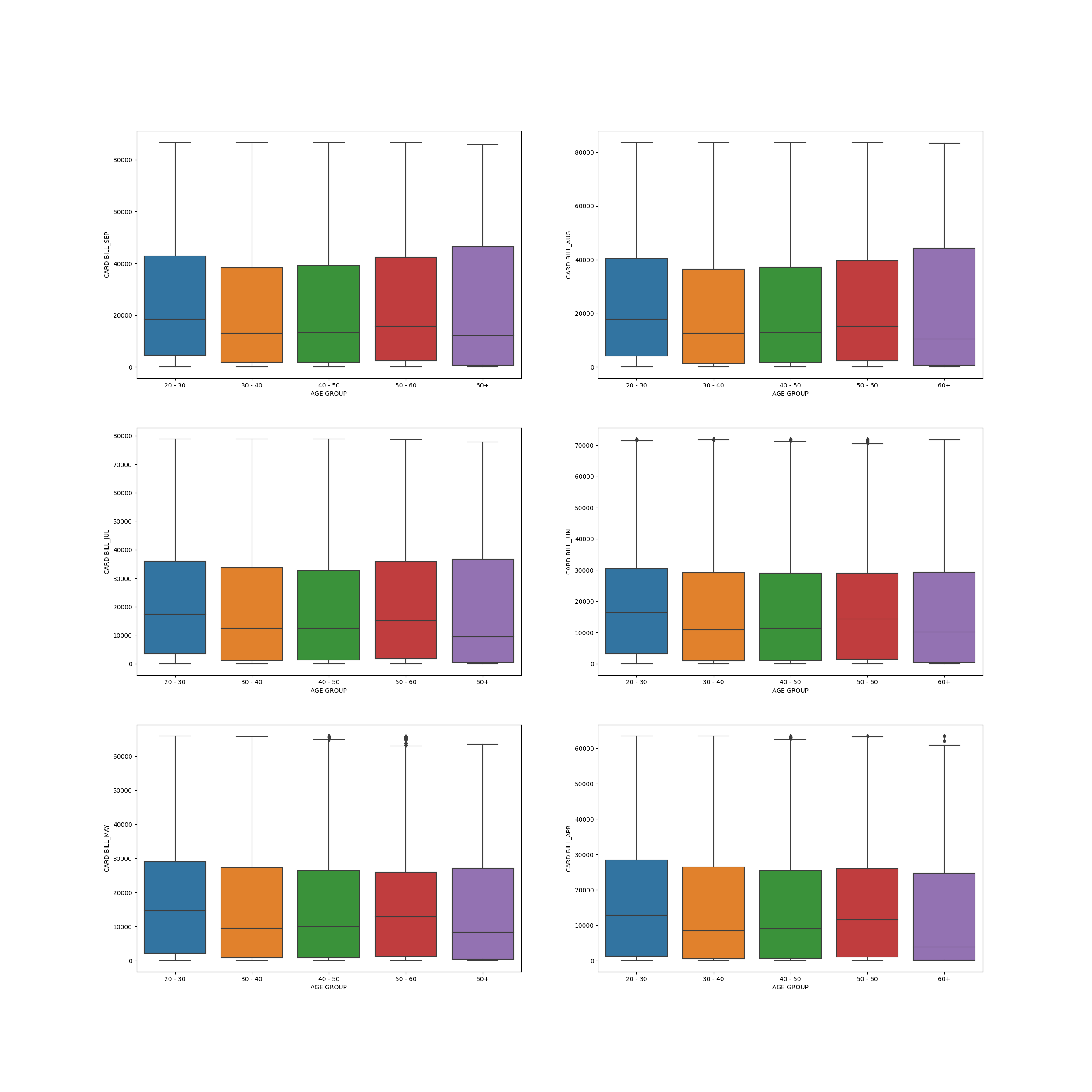
우선 신용한도는 평균적으로 30대가 가장 높았다. 그후 40대, 50대, 20대 순이었다. 60대 이상은 표본의 수가 부족하므로 신뢰성이 다소 떨어진다. 평균적으로 30대가 부여받는 신용한도는 40대보다 약 20%, 50대보다 약 63%, 20대보다 약 2배정도 높았다.





카드대금 상환 현황의 경우에는 비율상 20대가 가장 연체비율이 높았고 30대 이상은 서로 유사했다. 그래프로 살펴보면 대부분은 연체가 없었으며 20대는 시기에 따라 다르지만 평균적으로 약 20%가, 30대와 40대는 약 15% ~ 16%가, 50대는 약 10% ~ 20%가 연체가 있었다. 50대의 경우에는 시기에 따라 20대와 비슷한 경우가 있었지만 표본이 20대보다 적고 범위가 넓은 점을 고려하면 20대가 가장 연체 비율이 높고 나머지 연령대는 차이가 적다고 보는 것이 좋다.



CardBillVariable = ['CARD BILL\_SEP', 'CARD BILL\_AUG', 'CARD BILL\_JUL', 'CARD BILL\_JUN', 'CARD BILL\_MAY', 'CARD BILL\_APR']   
for i in range(len(CardBillVariable)):  
 Q1\_cb = dataframe[CardBillVariable[i]].quantile(0.25)  
 Q3\_cb = dataframe[CardBillVariable[i]].quantile(0.75)  
 IQR\_cb = Q3\_cb - Q1\_cb  
 dataframe\_CB = dataframe.loc[(dataframe[CardBillVariable[i]] > Q1\_cb - 0.25 \* IQR\_cb) & (dataframe[CardBillVariable[i]] < Q3\_cb + 0.25 \* IQR\_cb)]

print(dataframe.groupby(['AGE GROUP'])[CardBillVariable[i]].median())

>>> 20 - 30 25181.5

30 - 40 25718.0

40 - 50 25163.5

50 - 60 24980.5

60+ 27414.0

Name: CARD BILL\_SEP, dtype: float64

20 - 30 24220.5

30 - 40 24781.0

40 - 50 23807.5

50 - 60 24666.0

60+ 25225.5

Name: CARD BILL\_AUG, dtype: float64

20 - 30 22494.0

30 - 40 23280.0

40 - 50 21437.0

50 - 60 22561.5

60+ 25304.5

Name: CARD BILL\_JUL, dtype: float64

20 - 30 20100.5

30 - 40 21435.0

40 - 50 19433.5

50 - 60 19856.0

60+ 21922.0

Name: CARD BILL\_JUN, dtype: float64

20 - 30 19317.5

30 - 40 19933.0

40 - 50 18520.0

50 - 60 18882.5

60+ 20779.5

Name: CARD BILL\_MAY, dtype: float64

20 - 30 18852.0

30 - 40 19283.0

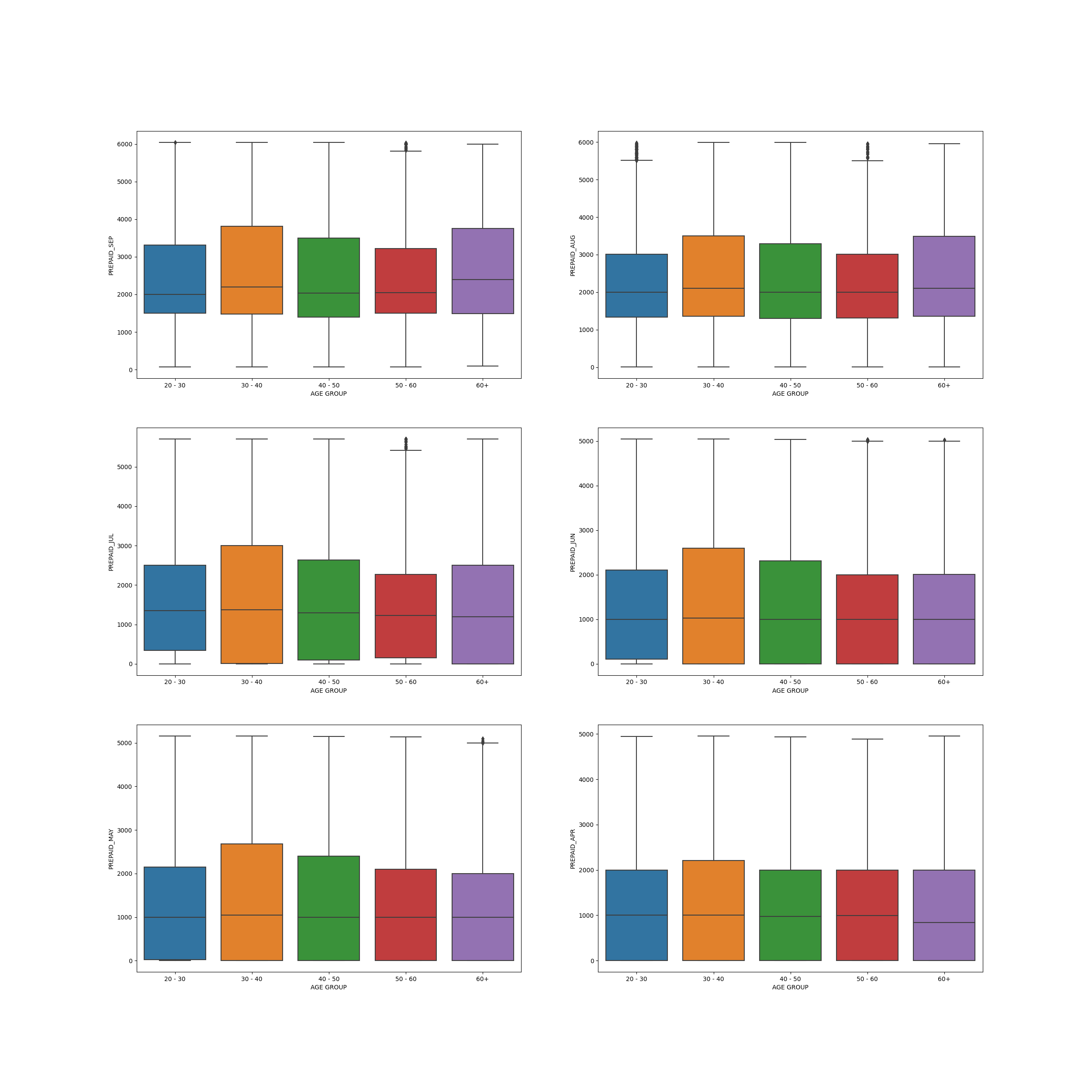
40 - 50 17514.0

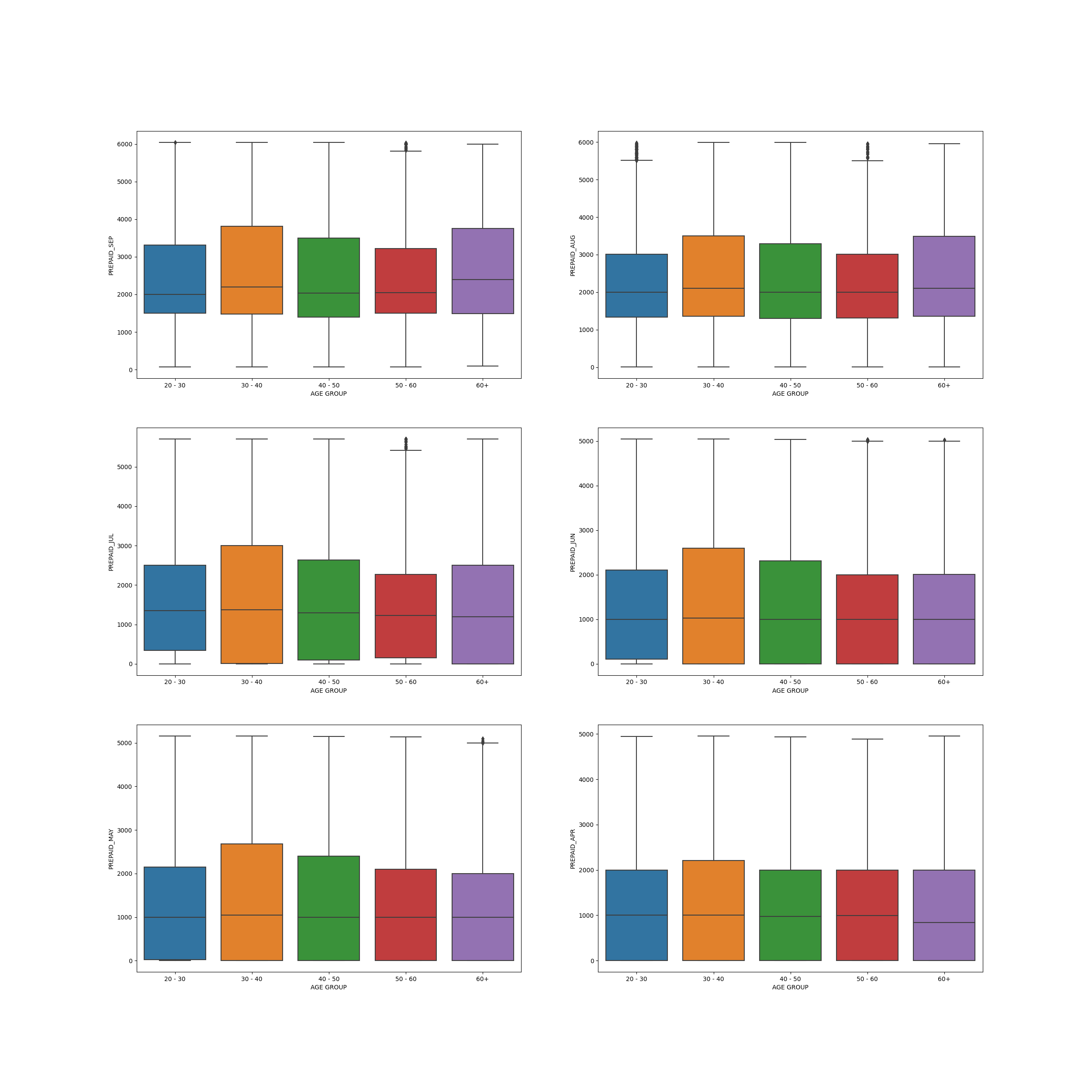
50 - 60 18683.5

60+ 19914.0

Name: CARD BILL\_APR, dtype: float64

신용카드 청구액의 경우에는 그래프상에서 20대가 가장 높았으며 그 다음이 50대였고 30대와 40대는 가장 낮았다. 다만 이상치를 포함하면 이야기가 달라졌다. 30대가 가장 높았고 그 다음이 20대와 50대였으며 40대는 20대와 비슷하거나 낮았다. 30대가 일반적으로는 20대보다 신용카드로 덜 소비하지만 많이 소비하는 상위권 고객은 20대와 비교하더라도 상당한 액수의 금액을 신용카드로 소비한다는 것을 의마한다. 각 연령대별 청구액 차이는 이상치를 포함했을 때 10% 이하였다.





CardBillVariable = ['CARD BILL\_SEP', 'CARD BILL\_AUG', 'CARD BILL\_JUL', 'CARD BILL\_JUN', 'CARD BILL\_MAY', 'CARD BILL\_APR']   
for i in range(len(CardBillVariable)):  
 Q1\_cb = dataframe[CardBillVariable[i]].quantile(0.25)  
 Q3\_cb = dataframe[CardBillVariable[i]].quantile(0.75)  
 IQR\_cb = Q3\_cb - Q1\_cb  
 dataframe\_CB = dataframe.loc[(dataframe[CardBillVariable[i]] > Q1\_cb - 0.25 \* IQR\_cb) & (dataframe[CardBillVariable[i]] < Q3\_cb + 0.25 \* IQR\_cb)]

print(dataframe.groupby(['AGE GROUP'])[CardBillVariable[i]].median())

>>> 20 - 30 2000.0

30 - 40 2620.0

40 - 50 2300.0

50 - 60 2100.5

60+ 2106.0

Name: PREPAID\_SEP, dtype: float64

20 - 30 2000.0

30 - 40 2532.0

40 - 50 2127.0

50 - 60 2000.0

60+ 2068.5

Name: PREPAID\_AUG, dtype: float64

20 - 30 1630.0

30 - 40 2000.0

40 - 50 2000.0

50 - 60 1695.0

60+ 1748.0

Name: PREPAID\_JUL, dtype: float64

AGE GROUP

20 - 30 1349.5

30 - 40 2000.0

40 - 50 1524.5

50 - 60 1397.5

60+ 1634.0

Name: PREPAID\_JUN, dtype: float64

20 - 30 1400.0

30 - 40 2000.0

40 - 50 1701.5

50 - 60 1443.0

60+ 1352.5

Name: PREPAID\_MAY, dtype: float64

20 - 30 1300.0

30 - 40 2000.0

40 - 50 1581.5

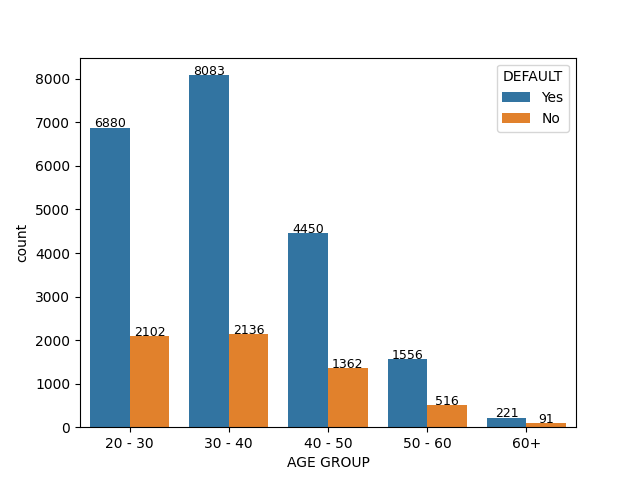
50 - 60 1316.0

60+ 1301.5

Name: PREPAID\_APR, dtype: float64

선불결제 이용액의 경우에는 그래프상에서 60대를 제외하고 30대가 가장 높았으며 나머지 연령대에서는 비슷한 수준이었다. 다만 그 차이는 그지 않았다. 그러나 Q3 값에서는 30대가 눈에 띄게 높았으며 그 다음이 40대였고 50대와 20대는 가장 낮았다. 중앙값보다 연령대에 따라 차이가 컸다. 이상치를 포함할때 역시 30대가 가장 높았으나 40대가 경우에 따라 30대와 비등한 경우도 많았다. 20대와 50대는 3040보다 청구액이 눈에 띄게 낮았다. 이때 30대는 20대나 50대보다 최대 약 53% 높았고, 40대보다 최대 33% 최소 0% 높았다.

이는 평균적인 고객들은 이용액이 어느정도 비슷하지만 상위권 사용자를 비교하면 30대와 40대의 소비가 많다는 것을 의미한다.



채무 불이행 가능성의 경우 그래프상에서 20대 26%, 30대 약 21%, 40대 약 23%, 50대 약 25%였다. 20대와 50대가 비슷하게 가장 높았고 40대가 중간정도였으며 30대가 가장 낮았다. 20대와 30대의 절대치 차이는 약 5%p, 상대적인 차이는 약 23% 였다.

정리하면 30대와 40대에게는 신용한도가 비교적 높게 부여되고 20대와 50대에게는 신용한도가 비교적 낮게 부여되고 있다. 사용패턴을 살펴보면, 30대는 신용카드와 선불결제 모두에서 상위권 고객의 소비력이 높고 연체 가능성도 가장 낮았다. 따라서 카드사라면 30대에게는 높은 한도를 부여함과 동시에 프리미엄 카드, 선불카드 가입을 추천하는 것이 좋다. 40대는 30대 다음으로 연체 가능성이 낮았고 선불카드에서 상위권 고객의 소비가 많았다. 다만 신용카드에서는 차이가 크지는 않으나 소비력이 상대적으로 떨어졌다. 따라서 선불카드의 가입을 추천함과 동시에 30대보다는 약간 낮은 한도를 부여하는 것이 옳다. 20대와 50대는 신용카드와 선불카드 모두에서 소비를 많이 하지 않음에도 불구하고 연체 가능성은 가장 높았다. 따라서 상대적으로 낮은 한도를 부여하는 것이 좋다.

**분석의 보완점** – 정밀한 이상치 제거 방법 필요함, 이상치를 포함한 데이터와 제외한 데이터를 모두 비교할 필요있음, 관측값의 수치 시각화를 더 간단하고 직관으로 할 방법을 찾을 필요 있음, 시각화된 자료 해석에 정확한 통계적 기법을 사용할 필요있음